UNIVERSITEIT
GENT
Faculteit Wetenschappen

Colour Morphology with Application to Image Magnification

Kleurenmorfologie met toepassing op het vergroten van beelden

Valérie De Witte
Licentiaat in de Wiskunde, optie Zuivere Wiskunde

Promotor:
Prof. Dr. E.E. Kerre

Proefschrift ingediend tot het behalen van de graad van
Doctor in de Wetenschappen: Wiskunde

Vakgroep Toegepaste Wiskunde en Informatica
Academiejaar 2006-2007

Dankwoord

Als eerste wil ik zeker en vast mijn promotor professor Kerre bedanken voor de kans dat ik dit onderzoek mocht verrichten, voor de steun en begeleiding bij het verwezenlijken van mijn doctoraat, voor de leuke tijd die ik hier heb beleefd en om steeds in mij te vertrouwen.
Ook mijn collega's binnen de onderzoeksgroep Vaagheids- en Onzekerheidsmodellering ben ik zeer dankbaar. Annelies, bedankt om mij te helpen bij de begeleiding van het practicum wiskunde, de oefeningen vaagheid en mijn onderzoek, en voor je toffe vriendschap. Bart, Klaas, Steven, Timur en Yun, bedankt voor de onvergetelijke momenten samen. Chris, dank je dat ik altijd bij je welkom ben. Dietrich, bedankt voor je steun, je interesse in alles wat ik doe en de mooie vriendschap die we opgebouwd hebben. Glad, bedankt om me telkens opnieuw te helpen wanneer ik je bureau binnenviel met een Tex-probleem en voor de leerrijke discussies die we samen gevoerd hebben. Martine, bedankt dat ik altijd bij jou terecht kan, bedankt om me te steunen en altijd in mij te geloven. Je enthousiasme geeft me energie! Mike, bedankt om me hier meteen thuis te laten voelen als ik begon, en altijd voor me op te komen. Patricia, ik weet dat ik je te veel plaag en bij jou niet altijd de juiste woorden vind, maar bij deze wil ik je oprecht bedanken voor je vriendschap, de ontelbare babbeltjes samen en om steeds een luisterend oor voor mij te zijn. Tom, bedankt om een groot deel van mijn doctoraat zorgvuldig te lezen, de goede ideëen die je me daarbij gaf, en de plezante tijd die we samen op ons bureau delen. Als laatste in dit rijtje wil ik Stefan bedanken. Stefan, ik wil dat je weet dat ik je dankbaar ben voor alles wat je voor mij gedaan hebt en nog steeds voor me doet. Je bent gewoon een super vriend! Verder wil ik alle andere leden van de vakgroep Toegepaste Wiskunde en Informatica bedanken voor de aangename werksfeer en de vele plezierige momenten. En zo dank ik ook buiten onze vakgroep de collega's van de vakgroep Telecommunicatie en Informatieverwerking die aan het GOA project meewerken, in het bijzonder Alessandro en Hiep, voor de fijne samenwerking.
Dank aan mijn familie en vrienden, voor alles. Ik wil ook mijn vriend Hiep bedanken omdat ik altijd op hem kan rekenen en hij er altijd voor mij is. Dank je om van me te houden! Tot slot dank ik mijn ouders en mijn zusje :) voor de onvoorwaardelijke steun en liefde die ze me geven en om zo goed voor mij te zorgen!!

Contents

Chapter 1

Introduction

Mathematical morphology (MM) is a theory for the analysis of spatial structures, based on set-theoretical notions and on the concept of translation. In the sixties G. Matheron and J. Serra [41] introduced the concept of MM, both inspired by the study of the geometry of porous media. Porous media are binary in the sense that a point of a porous medium either belongs to a pore or to the matrix surrounding the pores.
Matheron and Serra developed a theory for analysing binary images. The matrix can be considered as the set of object points and the pores as the complement of this set. As a consequence, image objects can be processed with simple operations such as union, intersection, complement and translation. MM was thus originally developed for binary images only. And so Matheron and Serra gave the basis for MM as a new approach in image analysis. Nowadays MM has many applications in image analysis such as edge detection, noise removal, object recognition, pattern recognition, image segmentation and image magnification in a.o. geosciences, materials science, the biological and medical world [71, 74]. The basic tools of MM are the morphological operators, which transform an image A we want to analyse, using a structuring element B into a new image $P(A, B)$ in order to obtain additional information about the objects in A like shape, size, orientation, image measurements. Apart from the threshold and umbra approach, binary morphology can be extended to morphology for greyscale images using fuzzy set theory, called fuzzy morphology. The application of morphological operators to colour images is not straightforward. And this is what this work is concerned with.

In chapter 2 we begin with the representation of digital images and some definitions about fuzzy sets, fuzzy logical operators, $\mathcal{L}$-fuzzy sets, $\mathcal{L}$-fuzzy logical operators, $\mathcal{L}$-fuzzy relations and $\mathcal{L}$-fuzzy relational images we will need further in this thesis. In the third chapter we explain how the human eye is build and how the human eye perceives light and so colour. We then describe additive and subtractive colour mixing to reproduce colour and give the difference between the terms colour model and colour space.

Lastly, we study the colour models RGB, CMY and CMYK, YUV, YIQ and YCbCr, HSV and HSL, CIEXYZ, CIEYxy, L*a*b* and L*u*v* in chapter 3. The definition of the fundamental morphological operators dilation and erosion is introduced in chapter 4. We consider the binary morphological operators as well as the greyscale morphological operators based on the threshold and umbra approach and on fuzzy logic. Next we give an overview and a short description of the existing extensions of MM to colour we found in the literature. A first way to apply the morphological operators for greyscale images to colour images is the component-based approach of processing the morphological operators on each of the colour components separately. But this approach often leads to artefacts because the connection between the colour components is not taken into account. Therefore we have searched for a vector ordering of colours, where we have considered the RGB, HSV and L*a*b* colour model. Subsequently, we have defined associated minimum and maximum operators and new operations $+$, $-$ and $*$ between colours so that we could extend the greyscale morphological operators to new vector-based operators acting on colour images. The problem of looking for a vector ordering for colour or multivariate morphological image processing is not new and is being developed since the early 90's. What is new here is the used approach, namely through the umbra approach and fuzzy set theory. Experimental results show that we get very good results. At last, in chapter 5 we have applied our new approach to colour morphology to magnify images. Image magnification has many applications such as simple spatial magnification of images (e.g. printing low-resolution documents on high-resolution printer devices, digital zoom in digital cameras), geometrical transformation (e.g. rotation), etc. Different image magnification methods have already been proposed in the literature, a.o. [2, 6, 20, 27, 32, 34, 38, 42, 54, 76, 78], but we have found only one paper [1] about techniques for the enlargement of images making use of MM. Because the existing methods usually suffer from one or more artefacts such as staircasing and blurring, we have developed a new image interpolation method, based on MM, to magnify images, binary images as well as colour images, with sharp edges. Whereas a simple blow up of the image will introduce jagged edges (staircasing effect), called 'jaggies', our method avoids these jaggies by detecting jagged edges in the trivial nearest neighbour interpolated image, making use of the hit-or-miss transformation, so that the edges become smoother. Experiments show that our method performs very well for the interpolation of 'sharp' images, like logos, cartoons and maps, for binary images as well as for colour images. Finally we propose an extension of our morphological interpolation method to magnify colour images with 'vague' edges. We demonstrated quite good results on this topic and an improvement w.r.t. the state-of-the-art.

Some parts of this thesis have already been published in a book [15] and in international journals [14, 16], and have been presented on (international) conferences [13, 14, 16]. Contributions to other people's work have been published in book chapters [69], international journals [30, 46, 48, 49, 60, 61, 62, 64, 65, 66, 68] and proceedings of (international) conferences [29, 31, 45, 47, 50, 63, 67, 82, 83, 84, 86, 85].

Chapter 2

Basic Notions

In this chapter we start with a brief explanation how digital images can be represented, and give some basic definitions from fuzzy set theory.

2.1 Representation of Images

A **digital image** I is represented by a two-dimensional array, where an ordered pair (i, j) denotes the position of a **pixel** or **picture element** $I(i, j)$ in the image I. The **resolution** of an image is the number of pixels per unit area, and is usually measured in pixels per inch. We distinguish three different kinds of digital images: binary, greyscale and colour images.

Binary images assume only two possible pixel values, e.g. 0 and 1, respectively corresponding to black and white. Usually white represents the foreground or the objects in an image, whereas black represents the background. Mathematically, a two-dimensional binary image can be represented as a mapping f from a universe U of pixels (usually U is a finite subset of $\mathbb{R}^2$, in practice it will even be a subset of $\mathbb{Z}^2$) into $\{0, 1\}$, which is completely determined by $f^{-1}(\{1\})$, i.e., the set of white pixels, so that f can be identified with the set $f^{-1}(\{1\})$, a subset of U, the so-called domain of the image. This way a two-dimensional binary image I can be represented as a crisp subset of U, with

$$\begin{array}{rcl} u \in I & \Leftrightarrow & u \text{ is a white pixel} \\ u \notin I & \Leftrightarrow & u \text{ is a black pixel,} \end{array}$$

for all u in U.

Greyscale images are images that contain, except black and white, also pixels with intermediate values between black and white, called grey values. A two-dimensional greyscale image I can be represented as a mapping from a universe U of pixels to the

universe of grey values $[0,1]$, where 0 corresponds to black, 1 to white and in between we have all shades of grey, where for every $u \in U$ holds that

$$\begin{array}{lcl} I(u) = 1 & \Leftrightarrow & u \text{ is a white pixel of } I \\ I(u) = 0 & \Leftrightarrow & u \text{ is a black pixel of } I \\ I(u) \in]0,1[& \Leftrightarrow & u \text{ is a grey pixel of } I. \end{array}$$

Colour images are represented as mappings from a universe U of pixels to a 'colour interval' that can be for example the product interval $[0,1] \times [0,1] \times [0,1]$ (for the RGB colour model). So a digital colour image in RGB is represented as a two-dimensional array of (three-dimensional) vectors that defines the red, green and blue colour component for each pixel. Colour can be modelled in different colour models; more information about colour and colour models can be found in chapter 3.

2.2 Fuzzy Set Theory

An extensive study of fuzzy set theory can be found in [28]. For more information about $\mathcal{L}$-fuzzy set theory we also refer to [12].

2.2.1 Fuzzy Sets

Given a universe X and a (crisp) set A in X ($A \subseteq X$).
The set A can then be represented by a characteristic mapping

$$\begin{array}{rccll} k_A: & X & \rightarrow & \{0,1\} & \\ & x & \mapsto & 1 & \text{if } x \in A \\ & x & \mapsto & 0 & \text{if } x \notin A. \end{array}$$

This way $k_A(x)$ can be interpreted as the membership degree of x in the set A in X, for all $x \in X$. In this case there are only two possible membership degrees: $k_A(x) = 0$ if the element x does not belong to the set A and $k_A(x) = 1$ if x belongs to A. Instead of a sharp boundary, L. Zadeh introduced in 1965 [88] a gradual transition from non-membership to membership, allowing partial degrees of membership. Mathematically this idea is translated into a **fuzzy set**. A fuzzy set A in a universe X is characterised by a $X - [0,1]$ mapping, the so-called **membership function**,

$$\begin{array}{rccl} \chi_A: & X & \rightarrow & [0,1] \\ & x & \mapsto & \chi_A(x), \quad \forall x \in X, \end{array}$$

where for all x in X, $\chi_A(x)$ denotes the degree in which x belongs to the fuzzy set A. $\chi_A(x) = 0$ means that x does not belong to the fuzzy set A at all, $\chi_A(x) = 1$ means that x belongs to A perfectly and between those two extremes there is a gradual transition from non-membership to membership. Furthermore we will denote the membership degree $\chi_A(x)$ as $A(x)$, where we no longer make a distinction between fuzzy sets

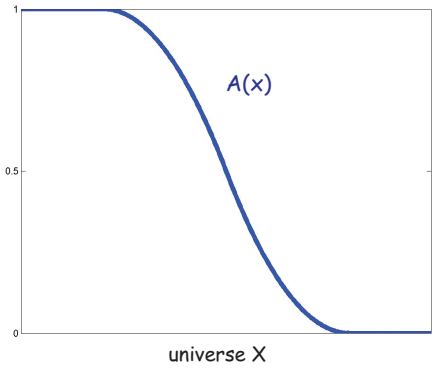


Figure 2.1: Graphical representation of a fuzzy set A in a universe X.

on the one hand and membership functions on the other hand. An example of a fuzzy set A in a universe X is shown in figure 2.1. The class of all fuzzy sets in X is denoted as $\mathcal{F}(X)$.

Consider $A_1 \in \mathcal{F}(X)$ and $A_2 \in \mathcal{F}(X)$. The **cartesian product** $A_1 \times A_2$ of A_1 and A_2 is defined as the fuzzy set

$$\begin{array}{rccl} A_1 \times A_2 : & X \times X & \rightarrow & [0,1] \\ & (x,y) & \mapsto & \min\{A_1(x), A_2(y)\}, \quad \forall (x,y) \in X \times X. \end{array}$$

The notion of cartesian product of two fuzzy sets in the same universe can be extended to a finite number of fuzzy sets. Let $A_i \in \mathcal{F}(X)$ with $i \in \{1, \ldots, n\}$, then

$$\begin{array}{rccl} A_1 \times \ldots \times A_n : & X^n & \rightarrow & [0,1] \\ & (x_1, \ldots, x_n) & \mapsto & \min\{A_i(x_i) \mid i \in \{1, \ldots, n\}\}, \end{array}$$

for all $(x_1, \ldots, x_n) \in X^n$.
When we consider two fuzzy sets in different universes $A_1 \in \mathcal{F}(X_1)$ and $A_2 \in \mathcal{F}(X_2)$, we define

$$\begin{array}{rccl} A_1 \times A_2 : & X_1 \times X_2 & \rightarrow & [0,1] \\ & (x_1, x_2) & \mapsto & \min\{A_1(x_1), A_2(x_2)\}, \quad \forall (x_1, x_2) \in X_1 \times X_2. \end{array}$$

2.2.2 Fuzzy Logical Operators

Now we will extend the logical operations negation ($\neg$), conjunction ($\wedge$), disjunction ($\vee$) and implication ($\Rightarrow$) to fuzzy logic, that is, to operators that apply to elements of the unit interval $[0,1]$ and that have a result in $[0,1]$. The restriction of these fuzzy logical operators to $\{0,1\}$ have to coincide with the corresponding two-valued operators.

In order to give a useful meaning on these operators further conditions concerning the monotonous character and some boundary conditions are required.

Definition 2.1. *A **conjunctor** $\mathcal{C}$ **on** $[0,1]$ is an increasing $[0,1]\times[0,1]-[0,1]$ mapping satisfying $\mathcal{C}(0,0)=\mathcal{C}(0,1)=\mathcal{C}(1,0)=0$ and $\mathcal{C}(1,1)=1$. A **triangular norm** (or **t-norm**) $\mathcal{T}$ **on** $[0,1]$ is a commutative and associative conjunctor on $[0,1]$ satisfying $\mathcal{T}(1,a)=\mathcal{T}(a,1)=a,\ \forall a\in[0,1]$.*

Definition 2.2. *A **negator** $\mathcal{N}$ **on** $[0,1]$ is a decreasing $[0,1]-[0,1]$ mapping satisfying $\mathcal{N}(0)=1$ and $\mathcal{N}(1)=0$. An **involutive negator** $\mathcal{N}$ **on** $[0,1]$ is a negator that satisfies the extra condition $\mathcal{N}(\mathcal{N}(a))=a$ for all a in $[0,1]$.*

Definition 2.3. *A **disjunctor** $\mathcal{D}$ **on** $[0,1]$ is an increasing $[0,1]\times[0,1]-[0,1]$ mapping satisfying $\mathcal{D}(1,0)=\mathcal{D}(0,1)=\mathcal{D}(1,1)=1$ and $\mathcal{D}(0,0)=0$. A **triangular conorm** (or **t-conorm**) $\mathcal{S}$ **on** $[0,1]$ is a commutative and associative disjunctor on $[0,1]$ satisfying $\mathcal{S}(0,a)=\mathcal{S}(a,0)=a,\ \forall a\in[0,1]$.*

Definition 2.4. *An **implicator** $\mathcal{I}$ **on** $[0,1]$ is a $[0,1]\times[0,1]-[0,1]$ mapping satisfying $\mathcal{I}(0,0)=\mathcal{I}(0,1)=\mathcal{I}(1,1)=1$ and $\mathcal{I}(1,0)=0$, and what is more, $\mathcal{I}$ is decreasing in its first, and increasing in its second component.*

The standard negator $\mathcal{N}_s$ on $[0,1]$, defined as $\mathcal{N}_s(a)=1-a$ for all a in $[0,1]$, is an involutive negator.
The best known conjunctors $\mathcal{C}$ on $[0,1]$ are the triangular norms $\mathcal{T}_M$ (minimum), $\mathcal{T}_P$ (algebraic product) and $\mathcal{T}_W$ (Lukasiewicz triangular norm) with

$$\begin{aligned}\mathcal{T}_M(a,b) &= \min(a,b),\\ \mathcal{T}_P(a,b) &= a\cdot b,\\ \mathcal{T}_W(a,b) &= \max(0,a+b-1),\quad \forall(a,b)\in[0,1]^2.\end{aligned}$$

A graphical representation of the t-norms $\mathcal{T}_M$, $\mathcal{T}_P$ and $\mathcal{T}_W$ is illustrated in figure 2.2. It is shown that $\mathcal{T}_W\le\mathcal{T}_P\le\mathcal{T}_M$.

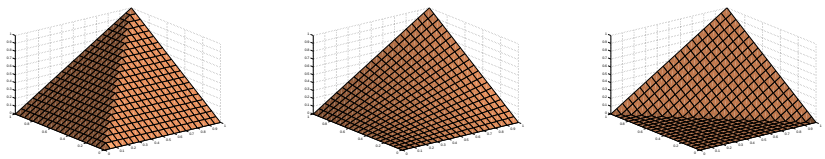


Figure 2.2: Graphical representation of the different t-norms, from left to right, the t-norm $\mathcal{T}_M$, the t-norm $\mathcal{T}_P$ and the t-norm $\mathcal{T}_W$.

The best known disjunctors $\mathcal{D}$ on $[0,1]$ are the triangular conorms $\mathcal{S}_M$ (maximum), $\mathcal{S}_P$

(probabilistic sum) and $\mathcal{S}_W$ (Lukasiewicz triangular conorm) with

$$\begin{aligned} \mathcal{S}_M(a,b) &= \max(a,b), \\ \mathcal{S}_P(a,b) &= a+b-a\cdot b, \\ \mathcal{S}_W(a,b) &= \min(1,a+b), \quad \forall (a,b)\in[0,1]^2. \end{aligned}$$

A graphical representation of the t-conorms $\mathcal{S}_M$, $\mathcal{S}_P$ and $\mathcal{S}_W$ is given in figure 2.3. It is shown that $\mathcal{S}_M \leq \mathcal{S}_P \leq S_W$.

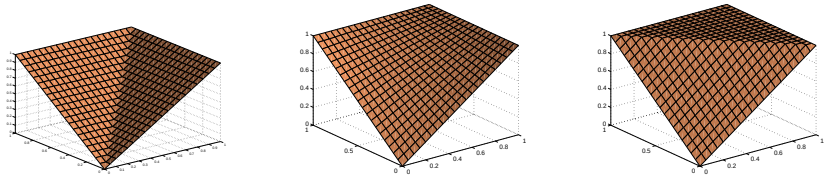


Figure 2.3: Graphical representation of the different t-conorms, from left to right, the t-conorm $\mathcal{S}_M$, the t-conorm $\mathcal{S}_P$ and the t-conorm $\mathcal{S}_W$.

The best known implicators $\mathcal{I}$ on $[0,1]$ are the Kleene-Dienes implicator $\mathcal{I}_{KD}$, the Reichenbach implicator $\mathcal{I}_R$ and the Lukasiewicz implicator $\mathcal{I}_W$ with

$$\begin{aligned} \mathcal{I}_{KD}(a,b) &= \max(1-a,b), \\ \mathcal{I}_R(a,b) &= 1-a+a\cdot b, \\ \mathcal{I}_W(a,b) &= \min(1,1-a+b), \quad \forall (a,b)\in[0,1]^2. \end{aligned}$$

A graphical representation of the implicators $\mathcal{I}_{KD}$, $\mathcal{I}_R$ and $\mathcal{I}_W$ is shown in figure 2.4. It holds that $\mathcal{I}_{KD} \leq \mathcal{I}_R \leq \mathcal{I}_W$.

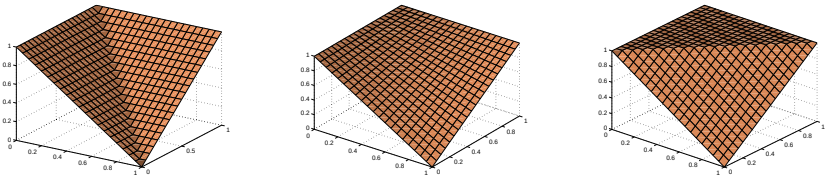


Figure 2.4: Graphical representation of the different implicators, from left to right, the implicator $\mathcal{I}_{KD}$, the implicator $\mathcal{I}_R$ and the implicator $\mathcal{I}_W$.

2.2.3 $\mathcal{L}$-Fuzzy Sets

Definition 2.5. *A binary relation $\leq$ over a set $\mathcal{P}$ is a **(partial) order relation** if and only if*

1. $(\forall a \in \mathcal{P})(a \leq a)$, *i.e.,* $\leq$ *is reflexive*
2. $(\forall (a,b) \in \mathcal{P}^2)(a \leq b \wedge b \leq a \Rightarrow a = b)$, *i.e.,* $\leq$ *is antisymmetric*
3. $(\forall (a,b,c) \in \mathcal{P}^3)(a \leq b \wedge b \leq c \Rightarrow a \leq c)$, *i.e.,* $\leq$ *is transitive.*

A set $\mathcal{P}$ *equipped with an order relation* $\leq$ *is called a* ***partially ordered set*** *(or* ***poset****), and noted as* $(\mathcal{P}, \leq)$. $a \geq b$ *stands for* $b \leq a$.

Definition 2.6. *A poset* $(\mathcal{P}, \leq)$ *satisfying* $(\forall (a,b) \in \mathcal{P}^2)(a \leq b \vee b \leq a)$*, i.e.,* $\leq$ *is total or linear, is called a* ***chain*** *or a* ***totally*** *(or* ***linearly) ordered set****. In a totally ordered set every two elements are comparable.*

Definition 2.7. *Let* $(\mathcal{P}, \leq)$ *be a poset,* $A \subseteq \mathcal{P}$ *and* $b \in \mathcal{P}$*. We then define*
b *is an* ***upper bound*** *of* A *if and only if* $(\forall a \in A)(a \leq b)$.
b *is a* ***lower bound*** *of* A *if and only if* $(\forall a \in A)(b \leq a)$.

A *is* ***bounded above*** $(in\ (\mathcal{P}, \leq))$ *iff* $(\exists b \in \mathcal{P})(b$ *is an upper bound of* $A)$.
A *is* ***bounded below*** $(in\ (\mathcal{P}, \leq))$ *iff* $(\exists b \in \mathcal{P})(b$ *is a lower bound of* $A)$.
A *is* ***bounded*** $(in\ (\mathcal{P}, \leq))$ *iff* A *is bounded above and bounded below* $(in\ (\mathcal{P}, \leq))$.

b *is the* ***greatest element*** *of* A *iff* $b \in A$ *and* b *is an upper bound of* A.
b *is the* ***smallest element*** *of* A *iff* $b \in A$ *and* b *is a lower bound of* A.
From the antisymmetric property one easily finds that the existence of a greatest element implies its uniqueness. If the greatest element exists, it will be denoted 1 *and the smallest element, if it exists, will be denoted* 0.

b *is the* ***supremum*** *of* A *iff* b *is the smallest upper bound of* A*;* $b = \sup(A)$.
b *is the* ***infimum*** *of* A *iff* b *is the greatest lower bound of* A*;* $b = \inf(A)$.

Definition 2.8. *A poset* $(\mathcal{L}, \leq)$ *is called a* ***lattice*** *if* $\inf(a,b)$ *and* $\sup(a,b)$ *exist for all* $a, b \in \mathcal{L}$*. We will use the notations* $a \wedge b$ *(*a *meet* b*) and* $a \vee b$ *(*a *join* b*) for* $\inf(a,b)$ *and* $\sup(a,b)$ *respectively.*

Definition 2.9. *A lattice* $(\mathcal{L}, \leq)$ *is* ***bounded*** *if there exist a greatest* (1) *and a smallest* (0) *element, i.e.,*

$$(\forall a \in \mathcal{L})(0 \leq a \leq 1)$$
or
$$(\forall a \in \mathcal{L})(1 \wedge a = a \ \text{ and } \ 0 \vee a = a).$$

Definition 2.10. *A lattice* $(\mathcal{L}, \leq)$ *is* ***complete*** *if every non-empty subset of* $\mathcal{L}$ *has a supremum and an infimum.*

Consider a bounded lattice $(\mathcal{L}, \leq_{\mathcal{L}})$ with smallest element denoted by 0 and greatest element by 1, join operator $\vee$ and meet operator $\wedge$. In 1967 Goguen [21] introduced **$\mathcal{L}$-fuzzy sets**, where the membership function has values in a lattice $(\mathcal{L}, \leq_{\mathcal{L}})$. An $\mathcal{L}$-fuzzy

set A in a universe X is characterised by a $X - \mathcal{L}$ mapping, called the **membership function of** A, and shortly denoted by A,

$$\begin{array}{rccl} A: & X & \rightarrow & \mathcal{L} \\ & x & \mapsto & A(x), \quad \forall x \in X. \end{array}$$

For all x in X, $A(x)$ expresses the membership degree of x in the $\mathcal{L}$-fuzzy set A, the degree in which x belongs to A. The class of all $\mathcal{L}$-fuzzy sets in X is denoted as $\mathcal{F}_{\mathcal{L}}(X)$. The fuzzy logical operators on $[0,1]$ can now be extended to $\mathcal{L}$-fuzzy logical operators.

2.2.4 $\mathcal{L}$-Fuzzy Logical Operators

Definition 2.11. *A **conjunctor** $\mathcal{C}$ **on** $\mathcal{L}$ is an increasing $\mathcal{L} \times \mathcal{L} - \mathcal{L}$ mapping satisfying $\mathcal{C}(0,0) = \mathcal{C}(0,1) = \mathcal{C}(1,0) = 0$ and $\mathcal{C}(1,1) = 1$. A **semi-norm** $\mathcal{C}$ **on** $\mathcal{L}$ is a conjunctor on $\mathcal{L}$ satisfying $(\forall a \in \mathcal{L})(\mathcal{C}(1,a) = \mathcal{C}(a,1) = a)$. A **triangular norm** (or **t-norm**) $\mathcal{T}$ **on** $\mathcal{L}$ is a commutative and associative conjunctor on $\mathcal{L}$ satisfying $\mathcal{T}(1,a) = a$ for all $a \in \mathcal{L}$.*

Definition 2.12. *A **negator** $\mathcal{N}$ **on** $\mathcal{L}$ is a decreasing $\mathcal{L}-\mathcal{L}$ mapping satisfying $\mathcal{N}(0) = 1$ and $\mathcal{N}(1) = 0$. $\mathcal{N}$ is called **involutive** if $\mathcal{N}(\mathcal{N}(a)) = a$ for all a in $\mathcal{L}$.*

Definition 2.13. *A **disjunctor** $\mathcal{D}$ **on** $\mathcal{L}$ is an increasing $\mathcal{L} \times \mathcal{L} - \mathcal{L}$ mapping satisfying $\mathcal{D}(1,0) = \mathcal{D}(0,1) = \mathcal{D}(1,1) = 1$ and $\mathcal{D}(0,0) = 0$. A **semi-conorm** $\mathcal{D}$ **on** $\mathcal{L}$ is a disjunctor on $\mathcal{L}$ satisfying $(\forall a \in \mathcal{L})(\mathcal{D}(0,a) = \mathcal{D}(a,0) = a)$. A **triangular conorm** (or **t-conorm**) $\mathcal{S}$ **on** $\mathcal{L}$ is a commutative and associative disjunctor on $\mathcal{L}$ satisfying $\mathcal{S}(0,a) = a$ for all $a \in \mathcal{L}$.*

Definition 2.14. *An **implicator** $\mathcal{I}$ **on** $\mathcal{L}$ is an $\mathcal{L}^2 - \mathcal{L}$ mapping satisfying $\mathcal{I}(0,0) = \mathcal{I}(0,1) = 1$ and $\mathcal{I}(1,a) = a$ for all a in $\mathcal{L}$. Moreover $\mathcal{I}$ is decreasing in its first, and increasing in its second component. An implicator $\mathcal{I}$ on $\mathcal{L}$ is an **edge-implicator** if $(\forall a \in \mathcal{L})(\mathcal{I}(1,a) = a)$.*

Property 2.15. *Let $\mathcal{C}$ be a conjunctor on $\mathcal{L}$. It holds:*

$$(\forall a \in \mathcal{L})(\mathcal{C}(0,a) = \mathcal{C}(a,0) = 0).$$

Proof
$\mathcal{C}(0,0) = \mathcal{C}(0,1) = \mathcal{C}(1,0) = 0$ *and $\mathcal{C}$ is increasing in its first and second component.*

■

Property 2.16. *Let $\mathcal{I}$ be an implicator on $\mathcal{L}$. It holds:*

$$(\forall a \in \mathcal{L})(\mathcal{I}(0,a) = \mathcal{I}(a,1) = 1).$$

Proof
$\mathcal{I}(0,0) = \mathcal{I}(0,1) = 1$ *and* $\mathcal{I}(1,a) = a$ *for all a in $\mathcal{L}$, and $\mathcal{I}$ is decreasing in its first and increasing in its second component.*

■

If $\mathcal{T}$ is a t-norm on $\mathcal{L}$, the mapping $\mathcal{I}_{\mathcal{T}}$ defined, for all a and b in $\mathcal{L}$, by

$$\mathcal{I}_{\mathcal{T}}(a,b) = \sup\{\lambda \mid \lambda \in \mathcal{L} \text{ and } \mathcal{T}(a,\lambda) \leq_{\mathcal{L}} b\}$$

is an implicator, called the **R-implicator of** $\mathcal{T}$. If the t-norm $\mathcal{T}$ is left-continuous, then we call the R-implicator $\mathcal{I}_{\mathcal{T}}$ of $\mathcal{T}$ a **residual implicator**. If $\mathcal{S}$ is a t-conorm and $\mathcal{N}$ a negator on $\mathcal{L}$, the mapping $\mathcal{I}_{\mathcal{S},\mathcal{N}}$ defined by

$$\mathcal{I}_{\mathcal{S},\mathcal{N}}(a,b) = \mathcal{S}(\mathcal{N}(a),b),$$

for all a and b in $\mathcal{L}$, is an implicator, called the $\mathcal{S}$**-implicator induced by** $\mathcal{S}$ **and** $\mathcal{N}$. If $\mathcal{T}$ is a t-norm and $\mathcal{N}$ an involutive negator on $\mathcal{L}$, the mapping $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ defined by

$$\mathcal{I}_{\mathcal{T},\mathcal{N}}(a,b) = \mathcal{N}(\mathcal{T}(a,\mathcal{N}(b))),$$

for all a and b in $\mathcal{L}$, is an implicator, called the $\mathcal{S}$**-implicator induced by** $\mathcal{T}$ **and** $\mathcal{N}$. We illustrate this with some examples on the unit interval: the three well-known t-norms and t-conorms on the lattice $([0,1],\leq)$ are $\mathcal{T}_M$, $\mathcal{T}_P$ and $\mathcal{T}_W$, and $\mathcal{S}_M$, $\mathcal{S}_P$ and $\mathcal{S}_W$ respectively. The residual implicators of $\mathcal{T}_M$, $\mathcal{T}_P$ and $\mathcal{T}_W$ are given by

$$\begin{aligned}
\mathcal{I}_{\mathcal{T}_M}(a,b) &= \begin{cases} 1 & \text{if } a \leq b \\ b & \text{otherwise} \end{cases}, \\
\mathcal{I}_{\mathcal{T}_P}(a,b) &= \begin{cases} 1 & \text{if } a \leq b \\ \frac{b}{a} & \text{otherwise} \end{cases}, \\
\mathcal{I}_{\mathcal{T}_W}(a,b) &= \min(1-a+b,1), \quad \forall (a,b) \in [0,1]^2;
\end{aligned}$$

and the $\mathcal{S}$-implicators induced by $\mathcal{T}_M$, $\mathcal{T}_P$ and $\mathcal{T}_W$ (and $\mathcal{S}_M$, $\mathcal{S}_P$ and $\mathcal{S}_W$) and the standard negator $\mathcal{N}_S$ are

$$\begin{aligned}
\mathcal{I}_{\mathcal{T}_M,\mathcal{N}_S}(a,b) &= \max(1-a,b), \\
\mathcal{I}_{\mathcal{T}_P,\mathcal{N}_S}(a,b) &= 1-a+a\cdot b, \\
\mathcal{I}_{\mathcal{T}_W,\mathcal{N}_S}(a,b) &= \min(1-a+b,1), \quad \forall (a,b) \in [0,1]^2.
\end{aligned}$$

So $(\mathcal{T}_M,\mathcal{I}_{KD}), (\mathcal{T}_P,\mathcal{I}_R)$ and $(\mathcal{T}_W,\mathcal{I}_W)$ are induced conjunctors and implicators on $[0,1]$.

Definition 2.17 (Complement, intersection, union). *Let $\mathcal{L}$ be a bounded lattice, $\mathcal{N}$ a negator, $\mathcal{T}$ a t-norm and $\mathcal{S}$ a t-conorm on $\mathcal{L}$. If A and B are $\mathcal{L}$-fuzzy sets in X, then $co_{\mathcal{N}}A$, $A \cap_{\mathcal{T}} B$ and $A \cup_{\mathcal{S}} B$ are $\mathcal{L}$-fuzzy sets in X, defined for all x in X as*

$$\begin{aligned}
co_{\mathcal{N}}A(x) &= \mathcal{N}(A(x)) \\
A \cap_{\mathcal{T}} B(x) &= \mathcal{T}(A(x),B(x)) \\
A \cup_{\mathcal{S}} B(x) &= \mathcal{S}(A(x),B(x)).
\end{aligned}$$

The Zadeh-intersection $A \cap_{\mathcal{T}_M} B$ and Zadeh-union $A \cup_{\mathcal{S}_M} B$ are noted as $A \cap B$ and $A \cup B$. For $co_{\mathcal{N}_s} A$ we shortly write coA.

For a family $(A_i)_{i \in I}$ of $\mathcal{L}$-fuzzy sets in X we define

$$\begin{aligned} \bigcap_{i \in I} A_i(x) &= \inf_{i \in I} A_i(x) \\ \bigcup_{i \in I} A_i(x) &= \sup_{i \in I} A_i(x) \end{aligned}$$

for all x in X.

It holds that ($\mathcal{F}_{\mathcal{L}}(X), \cap, \cup$) is a lattice with the ordering defined for all A and B in $\mathcal{F}_{\mathcal{L}}(X)$ as

$$A \subseteq B \Leftrightarrow (\forall x \in X)(A(x) \leq_{\mathcal{L}} B(x)).$$

2.2.5 $\mathcal{L}$-Fuzzy Relations and $\mathcal{L}$-Fuzzy Relational Images

Definition 2.18 ($\mathcal{L}$-fuzzy relations). *Let X and Y be two universes. An $\mathcal{L}$-**fuzzy relation** R **from** X **to** Y is an $\mathcal{L}$-fuzzy set on $X \times Y$, i.e., a $(X \times Y) - \mathcal{L}$ mapping. An $\mathcal{L}$-**fuzzy relation** R **on** X is an $\mathcal{L}$-fuzzy relation from X to X.*

Definition 2.19. *Let R be an $\mathcal{L}$-fuzzy relation from X to Y. For $x \in X$ we define the R-**afterset** of x as the $\mathcal{L}$-fuzzy set*

$$\begin{aligned} xR: \quad Y &\to \mathcal{L} \\ y &\mapsto R(x,y), \quad \forall y \in Y. \end{aligned}$$

*For $y \in Y$ we define the R-**foreset** of y as the $\mathcal{L}$-fuzzy set*

$$\begin{aligned} Ry: \quad X &\to \mathcal{L} \\ x &\mapsto R(x,y), \quad \forall x \in X. \end{aligned}$$

Definition 2.20 (Relational images). *Let R be a relation from a universe X to a universe Y and A a (crisp) subset of X. The **direct image of** A **under** R is given by*

$$R \uparrow A = \{y \mid y \in Y \wedge (\exists x \in X)\,(x \in A \wedge (x,y) \in R)\}\,.$$

The direct image of A is a subset of Y that contains all elements of Y that are connected with (at least) one element of A. So we can also write (in terms of R-foresets)

$$R \uparrow A = \{y \mid y \in Y \wedge A \cap Ry \neq \emptyset\}\,.$$

*The **superdirect image** $R \downarrow A$ **of** A **under** R is defined as the set of all elements of Y that are connected with all elements of A, thus*

$$R \downarrow A = \{y \mid y \in Y \wedge Ry \subseteq A\},$$

or

$$R \downarrow A = \{y \mid y \in Y \wedge (\forall x \in X)\,((x,y) \in R \Rightarrow x \in A)\}.$$

The definition of images of crisp sets under crisp relations can be extended to images of $\mathcal{L}$-fuzzy sets under $\mathcal{L}$-fuzzy relations.

Definition 2.21 ($\mathcal{L}$-fuzzy relational images). *Let $\mathcal{L}$ be a complete lattice, and X and Y two universes. For a triangular norm $\mathcal{T}$ on $\mathcal{L}$ and an implicator $\mathcal{I}$ on $\mathcal{L}$, an $\mathcal{L}$-fuzzy relation R from X to Y and an $\mathcal{L}$-fuzzy set A in X, the **direct image** $R \uparrow_{\mathcal{L}} A$ and the **subdirect image** $R \downarrow_{\mathcal{L}} A$ **of** A **under** R are the $\mathcal{L}$-fuzzy sets in Y defined as*

$$\begin{aligned} R \uparrow_{\mathcal{L}} A\,(y) &= \sup_{x \in X} \mathcal{T}(A(x), R(x,y)) \\ R \downarrow_{\mathcal{L}} A\,(y) &= \inf_{x \in X} \mathcal{I}(R(x,y), A(x)), \end{aligned}$$

for all y in Y.

We will normally make use of the simple notations above for direct and subdirect image. But when we work with more than one triangular norm or more than one implicator on $\mathcal{L}$ at the same time, we will make a distinction by adding the symbol of the triangular norm and implicator respectively in the notations of the direct image and subdirect image respectively to show which logical operator is used at that moment.

Property 2.22 (Monotonicity). *[12] Let $\mathcal{L}$ be a complete lattice, $\mathcal{T}$, $\mathcal{T}_1$ and $\mathcal{T}_2$ t-norms on $\mathcal{L}$ and $\mathcal{I}$, $\mathcal{I}_1$ and $\mathcal{I}_2$ implicators on $\mathcal{L}$. If R, R_1 and R_2 are $\mathcal{L}$-fuzzy relations from X to Y and A, A_1 and A_2 are $\mathcal{L}$-fuzzy sets in X, then it holds*

1. *If $R_1 \subseteq R_2$, then*
   - *(a)* $R_1 \uparrow_{\mathcal{L}} A \subseteq R_2 \uparrow_{\mathcal{L}} A$
   - *(b)* $R_1 \downarrow_{\mathcal{L}} A \supseteq R_2 \downarrow_{\mathcal{L}} A$
2. *If $A_1 \subseteq A_2$, then*
   - *(a)* $R \uparrow_{\mathcal{L}} A_1 \subseteq R \uparrow_{\mathcal{L}} A_2$
   - *(b)* $R \downarrow_{\mathcal{L}} A_1 \subseteq R \downarrow_{\mathcal{L}} A_2$
3. *If $\mathcal{T}_1 \subseteq \mathcal{T}_2$ and $\mathcal{I}_1 \subseteq \mathcal{I}_2$, then*
   - *(a)* $R \uparrow_{\mathcal{L},\mathcal{T}_1} A \subseteq R \uparrow_{\mathcal{L},\mathcal{T}_2} A$
   - *(b)* $R \downarrow_{\mathcal{L},\mathcal{I}_1} A \subseteq R \downarrow_{\mathcal{L},\mathcal{I}_2} A$

Property 2.23 (Interaction with $\mathcal{T}_M$-intersection of $\mathcal{L}$-fuzzy sets). *[12] Consider a complete lattice $\mathcal{L}$, a t-norm $\mathcal{T}$ on $\mathcal{L}$ and an implicator $\mathcal{I}$ on $\mathcal{L}$. If R is an $\mathcal{L}$-fuzzy relation in X and A and B are two $\mathcal{L}$-fuzzy sets in X, then it holds*

$$\begin{aligned} R \uparrow_{\mathcal{L}} (A \cap B) &\subseteq (R \uparrow_{\mathcal{L}} A) \cap (R \uparrow_{\mathcal{L}} B) \\ R \downarrow_{\mathcal{L}} (A \cap B) &\subseteq (R \downarrow_{\mathcal{L}} A) \cap (R \downarrow_{\mathcal{L}} B). \end{aligned}$$

Generalisation to a finite family $(A_i)_{i\in I}$ of $\mathcal{L}$-fuzzy sets in X leads to

$$\begin{aligned} R \uparrow_{\mathcal{L}} (\bigcap_{i\in I} A_i) &\subseteq \bigcap_{i\in I} (R \uparrow_{\mathcal{L}} A_i) \\ R \downarrow_{\mathcal{L}} (\bigcap_{i\in I} A_i) &\subseteq \bigcap_{i\in I} (R \downarrow_{\mathcal{L}} A_i). \end{aligned}$$

Property 2.24 (Interaction with $\mathcal{T}_M$-intersection of $\mathcal{L}$-fuzzy relations). *[12] Let $\mathcal{L}$ be a complete lattice, $\mathcal{T}$ a t-norm on $\mathcal{L}$ and $\mathcal{I}$ an implicator on $\mathcal{L}$. If R_1 and R_2 are $\mathcal{L}$-fuzzy relations from X to Y and A is an $\mathcal{L}$-fuzzy set in X, then it holds*

$$\begin{aligned} (R_1 \cap R_2) \uparrow_{\mathcal{L}} A &\subseteq (R_1 \uparrow_{\mathcal{L}} A) \cap (R_2 \uparrow_{\mathcal{L}} A) \\ (R_1 \cap R_2) \downarrow_{\mathcal{L}} A &\supseteq (R_1 \downarrow_{\mathcal{L}} A) \cup (R_2 \downarrow_{\mathcal{L}} A). \end{aligned}$$

For a finite family $(R_i)_{i\in I}$ of $\mathcal{L}$-fuzzy relations from X to Y we get

$$\begin{aligned} (\bigcap_{i\in I} R_i) \uparrow_{\mathcal{L}} A &\subseteq \bigcap_{i\in I} (R_i \uparrow_{\mathcal{L}} A) \\ (\bigcap_{i\in I} R_i) \downarrow_{\mathcal{L}} A &\supseteq \bigcup_{i\in I} (R_i \downarrow_{\mathcal{L}} A), \end{aligned}$$

and

$$(\bigcap_{i\in I} R_i) \downarrow_{\mathcal{L}} A \supseteq \bigcap_{i\in I} (R_i \downarrow_{\mathcal{L}} A).$$

Property 2.25 (Interaction with $\mathcal{S}_M$-union of $\mathcal{L}$-fuzzy sets). *[12] Consider a complete lattice $\mathcal{L}$, a t-norm $\mathcal{T}$ on $\mathcal{L}$ and an implicator $\mathcal{I}$ on $\mathcal{L}$. If R is an $\mathcal{L}$-fuzzy relation in X and A and B are $\mathcal{L}$-fuzzy sets in X, then it holds*

$$\begin{aligned} (R \uparrow_{\mathcal{L}} A) \cup (R \uparrow_{\mathcal{L}} B) &\subseteq R \uparrow_{\mathcal{L}} (A \cup B) \\ (R \downarrow_{\mathcal{L}} A) \cup (R \downarrow_{\mathcal{L}} B) &\subseteq R \downarrow_{\mathcal{L}} (A \cup B). \end{aligned}$$

Generalisation to a finite family $(A_i)_{i\in I}$ of $\mathcal{L}$-fuzzy sets in X leads to

$$\begin{aligned} \bigcup_{i\in I} (R \uparrow_{\mathcal{L}} A_i) &\subseteq R \uparrow_{\mathcal{L}} (\bigcup_{i\in I} A_i) \\ \bigcup_{i\in I} (R \downarrow_{\mathcal{L}} A_i) &\subseteq R \downarrow_{\mathcal{L}} (\bigcup_{i\in I} A_i). \end{aligned}$$

Property 2.26 (Interaction with $\mathcal{S}_M$-union of $\mathcal{L}$-fuzzy relations). *[12] Let $\mathcal{L}$ be a complete lattice, $\mathcal{T}$ a t-norm on $\mathcal{L}$ and $\mathcal{I}$ an implicator on $\mathcal{L}$. If R_1 and R_2 are $\mathcal{L}$-fuzzy relations from X to Y and A is an $\mathcal{L}$-fuzzy set in X, then it holds*

$$\begin{aligned} (R_1 \cup R_2) \uparrow_{\mathcal{L}} A &\supseteq (R_1 \uparrow_{\mathcal{L}} A) \cup (R_2 \uparrow_{\mathcal{L}} A) \\ (R_1 \cup R_2) \downarrow_{\mathcal{L}} A &\subseteq (R_1 \downarrow_{\mathcal{L}} A) \cap (R_2 \downarrow_{\mathcal{L}} A). \end{aligned}$$

For a finite family $(R_i)_{i\in I}$ of $\mathcal{L}$-fuzzy relations from X to Y we get

$$\begin{aligned}(\bigcup_{i\in I} R_i) \uparrow_{\mathcal{L}} A &\supseteq \bigcup_{i\in I} (R_i \uparrow_{\mathcal{L}} A)\\ (\bigcup_{i\in I} R_i) \downarrow_{\mathcal{L}} A &\subseteq \bigcap_{i\in I} (R_i \downarrow_{\mathcal{L}} A),\end{aligned}$$

and

$$(\bigcup_{i\in I} R_i) \downarrow_{\mathcal{L}} A \subseteq \bigcup_{i\in I} (R_i \downarrow_{\mathcal{L}} A).$$

Property 2.27 (Interaction with complement). *[12] Consider a complete lattice $\mathcal{L}$, a triangular norm $\mathcal{T}$ on $\mathcal{L}$, an involutive negator $\mathcal{N}$ on $\mathcal{L}$ and $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ the induced $\mathcal{S}$-implicator, then it holds for every $\mathcal{L}$-fuzzy relation R from X to Y and every $\mathcal{L}$-fuzzy set A in X that*

$$\begin{aligned}co_{\mathcal{N}}(R \uparrow_{\mathcal{L}} A) &= R \downarrow_{\mathcal{L}} (co_{\mathcal{N}} A)\\ R \uparrow_{\mathcal{L}} (co_{\mathcal{N}} A) &= co_{\mathcal{N}}(R \downarrow_{\mathcal{L}} A).\end{aligned}$$

Property 2.28 (Image of universe and empty set). *[12] Let $\mathcal{L}$ be a complete lattice, $\mathcal{T}$ a triangular norm on $\mathcal{L}$ and $\mathcal{I}$ an implicator on $\mathcal{L}$. For every $\mathcal{L}$-fuzzy relation R from X to Y it holds*

$$R \uparrow_{\mathcal{L}} \emptyset = \emptyset \quad \textit{and} \quad R \downarrow_{\mathcal{L}} X = Y.$$

Property 2.29 (Expansion and restriction). *[12] Let $\mathcal{L}$ be a complete lattice, $\mathcal{T}$ a t-norm on $\mathcal{L}$ and $\mathcal{I}$ an implicator on $\mathcal{L}$. If R is a reflexive $\mathcal{L}$-fuzzy relation in X, i.e., $(\forall x \in X)(R(x,x) = 1)$, and A an $\mathcal{L}$-fuzzy set in X, then*

$$R \downarrow_{\mathcal{L}} A \subseteq A \subseteq R \uparrow_{\mathcal{L}} A.$$

Chapter 3

Colour and Colour Models

In this chapter we first examine the structure of the human eye and study what colour really is to understand how the human eye perceives colour. We describe two methods to reproduce colour: additive and subtractive colour mixing. Finally we give a short summary of different colour models: RGB, CMY and CMYK, YUV, YIQ and YCbCr, HSV and HSL, CIEXYZ, CIEYxy, L*a*b* and L*u*v*.

3.1 Perception and Reproduction of Colour

3.1.1 Perception of Colour

The human eye

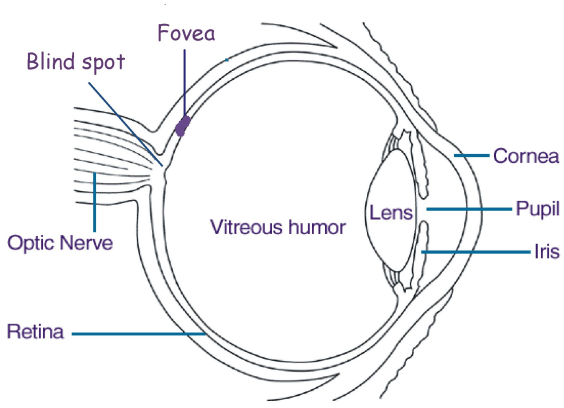


Figure 3.1: Image of the human eye.

In figure 3.1 [51] we see a sketch of the human eye. The **pupil** is an opening in the eye. All light that reaches the light sensitive parts of the eye has to pass through the pupil. The **iris** is a circular muscle that controls the size of the pupil. According to the size of the pupil, more or less light can enter the eye. By contracting and relaxing the iris, the eye can adapt to different light conditions like day and night.
The **cornea** is a protective, transparent layer that covers both the pupil and the iris, and it is the first lens of the human vision system. The **inner lens**, the second lens of the human eye, is a transparent body that is able to contract and relax. The shape of the inner lens can change and allows the eye to adapt to a large range of distances. This inner lens also focuses incoming light radiations, that is, all light coming from the same physical point of space is brought together (focused) in one single point on the **retina**. The retina consists of several layers with each a specific task, like for example receiving light or converting incoming light in electrical signals. The **fovea** is called the centre of the retina. This tiny area is responsible for our central, sharpest vision. Now, the retina contains two kinds of **light sensitive receptors**, also called **photoreceptors**, **cones** and **rods**, which 'translate' incoming light in electrical signals. The fovea contains no rods, but the compactness of the cones is largest in the fovea. The electrical signals produced by the cones and rods are then sent along the **optic nerve** to a part of the brain behind the head for further processing. The optic nerve fibres exit the eye in a small spot on the retina, called **blind spot**. The blind spot has no photoreceptors at all.

What is colour?

Light consists of electromagnetic waves. There is a complete spectrum of **electromagnetic radiation**, from radio waves to gamma rays.

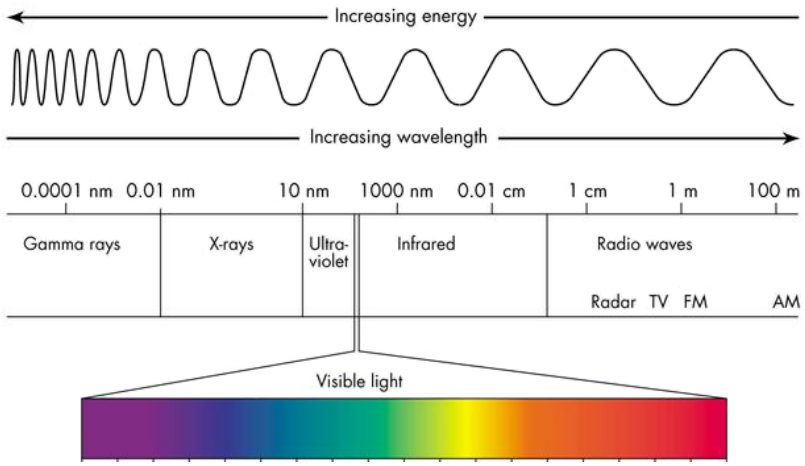


Figure 3.2: The electromagnetic spectrum.

The different kinds of electromagnetic radiation can be characterised by their frequency or by their wavelength, where both are connected by the equation

$$c = f.\lambda$$

with f the frequency expressed in s^{-1}, λ the wavelength expressed in nm ($1nm = 10^{-9}m$) and c the speed of light ($c = 3 \times 10^{8}ms^{-1}$). The human eye observes these different frequencies/wavelengths as different **colours**. Only a little part of the whole electromagnetic spectrum can be perceived by the human eye: visible colours for humans appear with a wavelength between about $400\ nm$ and $700\ nm$ in the electromagnetic spectrum, the so-called **visible light**. We call this part of the electromagnetic spectrum the **visible spectrum** too. In figure 3.2 [79] the electromagnetic spectrum is shown, going from radio waves with long wavelengths ($10^{7} - 10^{11}nm$) to the short gamma rays ($10^{-4}nm$).

As we have seen, the retina contains two different light receptors, cones and rods. Rods care for the dark vision, visibility in darkness, and are very sensitive for light. They only see in grey values, so they cannot distinguish colour, and cannot give us a sharp vision. Cones on the other hand are less sensitive for light than rods, but detect colour. They care for our sharp vision and visibility by daylight. There are three types of cones that are sensitive, especially for red, green and blue light: each cone type contains a light sensitive pigment that is sensitive over a range of wavelengths; the α-cones are sensitive in the blue part of the electromagnetic spectrum, the β-cones in the green part and the γ-cones in the yellow-green to red part. Note that because the compactness of the cones is largest in the fovea, this area of the retina has the highest visual sensitivity for colour perception.

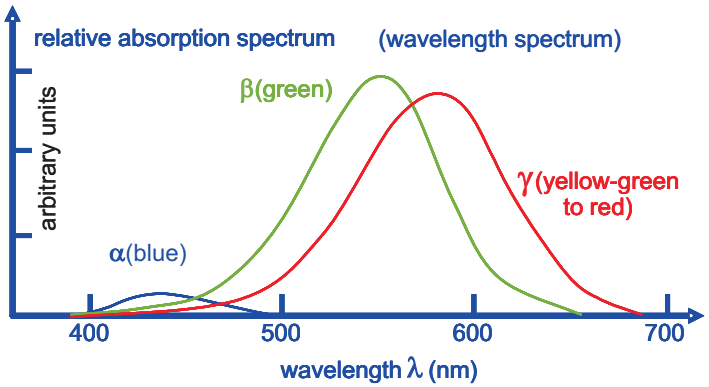


Figure 3.3: Absorption spectrum of the α-, β- and γ-cones.

Figure 3.3 [53] gives the absorption spectrum of the three types of cones, so we get an idea of the 'spectral sensitivity' of the cones, that is, how many light on a certain

wavelength is absorbed by the cones. We see that the α-cones are less sensitive than the other two types of cones and that one single wavelength can activate more than one type of cone.
Light that reaches a little environment on the retina that contains three kinds of cones is converted in three signals (one for each type of cone). The eye is **trichromatic**, i.e., an incoming light spectrum is seen as a point in a three-dimensional space. This trichromatic character of the eye is very important in image processing. It implies that each visible light point for humans (and thus every pixel in colour images) can be completely described by three numbers. The giant range of colours that can be seen by humans can be obtained by adding the right amounts of red, green and blue colour together or by adding two of these three colours together and subtracting the other (more information: see section 3.2.5). These signals coming from the cones are then sent to our brain where they are translated in a colour. When all wavelengths of the visible spectrum reach our eye at the same time, we see white. White is not a colour at all, but the mixture of all visible colours. That is why visible light is also called **white light**. Black is not a colour either, but corresponds with the total absence of wavelengths of the visible light spectrum.

We may conclude that the question 'what is colour?' is subjective. Colours are created in our brain as a reaction on light. Colour is how the human eye and brain observe different wavelengths of light.

Although the eye senses colour according to red, green and blue light (trichromacy), higher visual processes in the brain code colour according to the **opponent process theory**, using the opponent pairs red-green, blue-yellow and black-white. Humans perceive colour as having four distinct colour hues corresponding to the perceptually sensation of red, green, yellow and blue. We know that yellow can be produced by the additive mixing of red and green, but in our brain yellow is perceived as being qualitatively different from each of the two components red and green. The opponent process theory says that all colours can be described as containing red or containing green, but never as containing both red and green simultaneously. We can never see a colour that appears red and green at the same time, the colours red and green can never be perceived together. The same observation has been done for the colours blue and yellow. The opponent process theory also states that one cannot say this of any other pair of colours. Thus, classifying a colour as either red or green, and then independently as blue and yellow, gives a complete description of that colour.
The best way to arrange colours is in a wheel. The opponent process theory says that our sensation of colour is organised along two axes. The position along the first axis encodes the redness or greenness of the colour. The second independent axis encodes the blueness or yellowness of the colour. The complete colour sensation is the red-green and blue-yellow coordinates of that colour. Sometimes a third axis is added to the opponent colour wheel that describes the brightness (whiteness to blackness) of the colour. Opponent process theory also implies that there should be 'pure' colours: a red

and a green that has no blue or yellow in it, and a blue and a yellow that has no red or green in it. Only red, green, blue and yellow can be made pure. There is for example no pure orange, because orange always looks like a mix of red and yellow. This theory also explains the existence of some intermediate hues (e.g. red-yellow, yellow-green, green-blue and blue-red) and the absence of other intermediate hues (e.g. red-green and yellow-blue).
Simultaneous contrast also demonstrates a red-green and blue-yellow association. Simultaneous contrast is a colour appearance phenomenon that causes the 'colour appearance' of a colour element to shift when the colour of the background changes. This change in colour tends to follow the opponent colour theory. For example, a thin, grey, colourless line running over a red background appears slightly green, while running over a blue background it appears slightly yellow.

Colour blindness

We give a short description of colour blindness.
It is not exactly known how colour blindness arises. It is possible that by colour blind people one or more types of cones are not or in a smaller amount presented in the retina. It is also possible that all cones are presented in the retina, but that they function less good or totally not, or that the signals coming from the cones are not passed in the right way to the brain. Anyhow, one particular colour or more than one is perceived less intense whereby certain colours can no longer be distinguished. Red-green colour blindness is by far the most common form of colour blindness and causes problems in distinguishing reds and greens. Colour blindness is generally a hereditary disease. Colour blind people can see better in the dark.

There are many different types and degrees of colour blindness, called **colour deficiences**.

Monochromacy Monochromacy occurs in two forms: rod monochromacy (also called achromatopsia) and cone monochromacy.

- Rod monochromacy
  Only black, white and shades of grey can be seen. These colour blind people are truly colour blind because any colour cannot be perceived (complete absence of any colour sensation), and they even have difficulties with seeing in bright daylight. There can be three causes:
  - there are no cones at all in the retina, only rods
  - the cones have no or non-working colour sensitive photo pigments
  - the signals coming from the cones are not translated in the brain.

- Cone monochromacy
  Only one type of cones is active. These people see some colour, e.g. people with

blue cone monochromacy cannot perceive any colour except blue so that they have a feeling of seeing colour, but cannot distinguish colour hues too.

Dichromacy It is known that by dichromacy one of the three kinds of cones is missing and is replaced by one of the other two cone types. These colour blind people are able to match all colours using only two primaries rather than the normal three (trichromacy). Therefore they are blind to certain colour differences that normal individuals can see.

- Protanopia: red photo pigment is missing and is replaced by green photo pigment. This way it almost becomes impossible to make distinction between reds, yellows and greens. Short wavelengths are seen as blue. Red colours seem to be darker than for a normal viewer, and so that reds may be confused with black or dark grey. These people learn to distinguish reds from yellows and greens primarily on the basis of different degrees of brightness (lightness), not on any perceptible hue difference.

- Deuteranopia: green photo pigment is missing and is replaced by red photo pigment. Like by protanopia this leads to an inability in distinguishing between reds, yellows and greens. The brightness of red is preserved.

Because distinctions between red and green are based on the activity of the β- and γ-cones, both protanopia and deuteranopia are termed red-green colour blindness. There is almost no perceptible difference between red, orange, yellow and green: these colours appear to be the same colour. Both, protanopia and deuteranopia colour blind people see the world in shades of white, grey, black, blue and yellow.

- Tritanopia: blue photo pigment is missing and is replaced by red and green photo pigment. These colour blind people are insensitive to yellows and blues, but see the world in shades of white, grey, black, red and green.

Anomalous trichromacy All cones are active but there is a disturbed quantity of one or more of the three cones' colour sensitive pigments. Therefore the sensitivity for light has shifted, and so these people will mix the primaries in different proportions. For example, for a given spectral yellow light protanomalous observers will need more red light in a red-green mixture while deuteranomalous observers need more green than a normal observer.

- Protanomaly or red-weakness: abnormal red sensitivity (disturbed quantity of red-sensitive cones (γ-photo pigment)). Any redness seen in a colour by a normal observer is seen more weakly, both in terms of its colouring power (saturation) and brightness. Red, orange, yellow, yellow-green and green appear somewhat shifted in hue towards green, and all appear paler than for a normal observer. The redness component that a normal observer sees in a violet or lavender colour is

so weakened that maybe it would not be detected by a protanomalous observer, and so only the blue component will be seen. A colour that one calls 'violet' may look like another shade of blue.

- Deuteranomaly or green-weakness: abnormal green sensitivity (deviation of the quantity of green-sensitive cones (β-photo pigment)). Small differences in hues in the red, orange, yellow and green part of the visible spectrum are difficult to see: they appear somewhat shifted towards red, but there is no loss of brightness. Also violet, lavender, purple and blue appear as similar colours.
- Tritanomaly or blue-weakness: abnormal blue-yellow sensitivity (disturbed quantity of blue-sensitive cones (α-photo pigment)). For a tritanomalous observer it will be difficult to distinguish between yellow and blue.

3.1.2 Reproduction of Colour

Additive and subtractive colour mixing

Additive colour mixing is the method to create a new colour by adding two or three colours together, the **additive primaries** or **additive primary colours**. These primary colours are usually red, green and blue. Additive colours are produced by a combination of spectrum colours that are optical coloured by putting the colours very close to each other or by showing the colours very fast the one after the other so that the eye can no longer distinguish these colours at a normal view distance, but will mix or add them together to get a composed colour effect. This way the human eye observes two or more colours as one (new) colour.

Additive colour mixing works as follows:

1. Equal amounts of two primary colours in full intensity create a secondary colour:

$$\begin{aligned} 1\ \text{red} &+ 1\ \text{blue} = \text{magenta} \\ 1\ \text{blue} &+ 1\ \text{green} = \text{cyan} \\ 1\ \text{green} &+ 1\ \text{red} = \text{yellow}. \end{aligned}$$

2. Equal amounts of the three primary colours in full intensity create white:

$$1\ \text{red} + 1\ \text{blue} + 1\ \text{green} = \text{white}.$$

3. Unequal amounts of two or three primary colours (in full intensity or not) create different colours, for example,

$$\begin{aligned} 2\ \text{red} &+ 1\ \text{green} = \text{orange} \\ 1\ \text{red} &+ 2\ \text{green} = \text{lime}. \end{aligned}$$

   Other colours can be obtained in this way.

One of the most famous applications of additive colour mixing can be found in the television screen. The television screen is a mosaic that consists of thousands of tiny groups of phosphor dots. Each group contains three different kinds of phosphor dots: one kind of phosphor dots, the red phosphor dots, emits red light, the blue phosphor dots emit blue light and the green phosphor dots emit green light. The phosphor dots convert energy of the electrons on the cathode tube inside the television into radiation so that they can emit light. If there is no energy that the phosphor dots can convert, they do not radiate and then we observe the tiny group of phosphor dots as black. If the three kinds of phosphor dots (red, green and blue) in the tiny group radiate clearly all together at the same time, we observe the group as white. Different colours depend on how bright the red, green and blue phosphor dots radiate, e.g. we see cyan if both the blue and green phosphor dots radiate together. In figure 3.4 [70] a tiny group phosphor dots in the TV screen is illustrated.

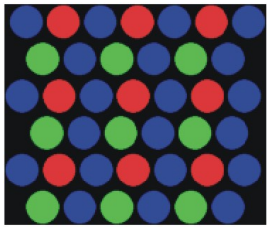


Figure 3.4: A tiny group phosphor dots in the television screen.

Subtractive colour mixing uses paints, pigments, inks or natural dyes to create colour by absorbing some wavelengths of white light and reflecting or transmitting others.
We see subtractive colours when pigments in an object absorb certain wavelengths of white light while reflecting or transmitting the rest. This happens all the time around us: when light reaches an object, the object absorbs some wavelengths of the light and reflects or transmits others. The wavelengths in the reflected or transmitted light make up the colour we see. As more ink is added, less and less light is reflected, which finally would be seen as black. When there is a total absence of ink, the light being reflected from a white surface is perceived as white. Consider as example a red apple. If white light reaches the red apple, all colours of the white light are absorbed by the surface of the apple except the colour red, which is reflected. The red light radiations reach our eyes and our brain observes the apple as a red apple. Another example: the leaves of green plants contain the pigment chlorophyll, which absorbs the blue and red colours of the spectrum and reflects the green colours. If an object absorbs all colours it receives, our brain will see the colour of that object as black.

This method forms the base for photographic filters, colour print productions like almost all films and colour paper, and photomechanical reproduction in colour. The pigments in colour filters and inks used for colour productions or in photomechanical

reproduction of colour are cyan, magenta and yellow.

Cyan, magenta, yellow are the complementary colours of red, green, blue and are called the **subtractive primaries** because of the following:
White light is composed of all wavelengths of the visible spectrum, which we can obtain by mixing equal amounts of the additive primaries red, green and blue in full intensity. A paint that absorbs one additive primary colour has the combined colour of the two other primaries. This combined colour is the complement of the absorbed colour abstracted of white light. We get

ABSORBED PRIMARY COLOUR	UNCHANGED PRIMARY COLOUR	COMBINED COLOUR
red	blue and green	cyan
green	blue and red	magenta
blue	red and green	yellow.

Now, cyan, magenta and yellow will subtract respectively red, green and blue of sunlight by absorbing this colour instead of reflecting it. Cyan, magenta and yellow together subtract all colour of white light, which results in black.

3.2 Colour Models and Colour Spaces

A **colour model** is an abstract mathematical model that describes how colours can be represented as tuples of numbers, usually as 3- or 4-tuples. A colour in a colour model can be specified by using coordinates or attributes. These coordinates do not tell us what the colour looks like, but represent where the colour is located in the particular colour model. The colours that can be represented using a particular colour model define then a **colour space**.

Colour models are used to specify, create and visualize colours so that they can be reproduced and processed in a clear way. Dependent on the application why colours are used for, different models have been developed for the characterisation and visualisation of colours. The set of all colours that can be produced on a device is called the **colour gamut**. Because of physical difference in how different devices produce colours, each scanner, display, printer, etc. has a different range of colours it can represent. There are colours that can be viewed on the screen but that cannot be reproduced in print, and conversely, some colours can be printed but not be seen on the screen. Colour output devices can be classified into three types: additive, subtractive and hybrid. Additive colour systems produce colour through the combination of differently coloured light. Colour in subtractive systems is produced through a process of removing (subtracting) unwanted spectral components from ‘white’ light. Subtractive colour

systems produce colour on transparent or reflective media, which have to be illuminated by white light for viewing. Hybrid colour systems use a combination of additive and subtractive processes to produce colour.

3.2.1 RGB Colour Model

The three additive primary colours red, green and blue form the base of the additive colour model RGB: a colour in RGB is obtained by adding the three colours red, green and blue in different combinations together. Every colour in the RGB colour model is totally determined by its three colour components R, G and B. Figure 3.5 shows the additive mixing of red, green and blue.

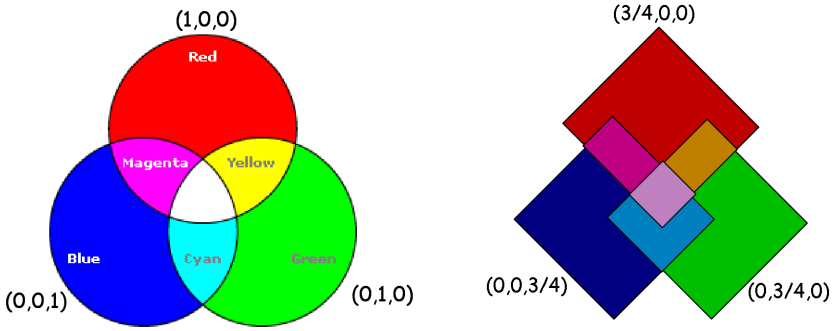


Figure 3.5: Additive mixing of the three RGB primary colours.

This way a colour can be defined as a vector in a three-dimensional space, which can be represented as a unit cube using a Cartesian coordinate scheme, see figure 3.6.
Every point in the cube (or vector with as starting point the origin) represents then a colour. Red, green and blue are the primary colours; cyan, magenta and yellow are the secondary colours. The greyscale spectrum, which consists of colours with the same amount of every RGB primary colour, lies on the line between the black and white top.

The colours that can be obtained in the RGB colour model are dependent on the way the colours ‘red’, ‘green’ and ‘blue’ are defined. So there exist different colour spaces based on the RGB colour model. Well-used RGB colour spaces are sRGB (standard RGB) and Adobe RGB. sRGB was developed by Hewlett-Packard and Microsoft Corporation using a simple and robust device independent colour definition to handle colour in operating systems, device drivers and the Internet. The Adobe RGB colour space was designed by Adobe Systems to encompass most of the colours achievable on CMYK colour printers, but by using RGB primary colours on a device such as the computer display.

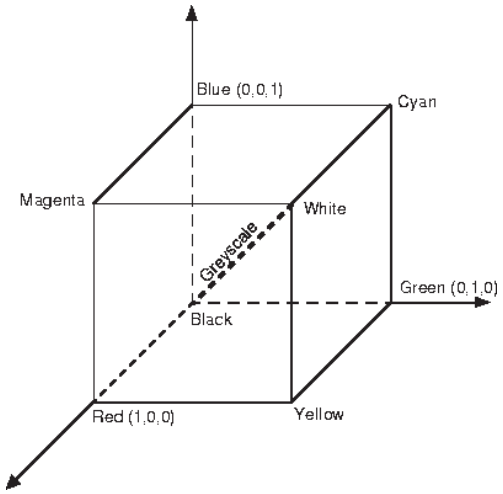


Figure 3.6: A representation of the RGB colour model expressed in Cartesian coordinates.

The RGB colour model has been developed because it is very close related to the manner how the human eye, with the α-, β- and γ-cones in the retina, observes colour. But in this model red, green and blue can only be added together and not subtracted, so that not all colours visible by the human eye can be obtained. This is a disadvantage.
The RGB colour model is used in television and computer displays, colour camera's and scanners.

3.2.2 CMY and CMYK Colour Model

The CMY colour model is a subtractive colour model, where a colour is described as a result of light being absorbed (subtracted) by printing inks with cyan, magenta and yellow as subtractive primary colours. In figure 3.7 we see the subtractive mixing of cyan, magenta and yellow to form red, green and blue and finally black by abstracting these three primaries from white.
The subtractive CMY and additive RGB colours are called **complementary colours**. Each pair subtractive colours creates an additive colour and conversely. That is why RGB and CMY are opposite colour models. Notice that the colours in the RGB colour model are brighter than those in the CMY colour model. In RGB a larger part of the visible spectrum can be reached. This is because the RGB model uses added light while the CMY model uses reflected light to create colours.

We have seen that pure cyan, magenta and yellow pigments together absorb all colour of white light and produce black. But practically it is impossible to create black with these three colours: because of impurities in inks not all coloured light will be absorbed

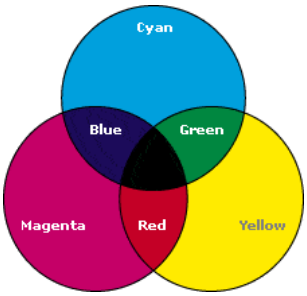


Figure 3.7: Subtractive mixing of the CMY primary colours.

by pure cyan, magenta and yellow pigments together so that these three colours do not produce black but a dark brown. That is why in practice black is added as fourth primary colour. We talk about **four-colour process printing** and call this the CMYK colour model.

The CMY and CMYK colour model are used in photographic colour filters, colour printers and photomechanical reproduction in colour.

Now we give some definitions related to light and colour we will need further on.

Hue is the attribute of a visual sensation according to which an area appears to be similar to one, or to the proportions of two, of the perceived colours red, yellow, green and blue. As a consequence of the opponent process theory we know that the colours red, yellow, green and blue are 'pure'. Black, white and the grey values in between are called 'neutral colours' because they have no hue.

The **saturation** of a colour indicates how much white light is presented in the colour. Red and pink for example are two different saturations of the same colour hue red. Pure colours are completely saturated, they contain no white light. If we add white to a pure colour, we get a lighter, less intense, desaturated colour. Pastel colours are less saturated because they are mixed with white. Colours that are not saturated are grey values. Saturation is sometimes called **colour intensity** too. In figure 3.8 [87] the change in saturation for the hue red is shown.

Figure 3.8: Change in saturation for the colour hue red, from 0% (right) to 100% (left).

Brightness is defined as the human visual sensation by which an area exhibits more or less light/appears to emit more or less light/seems to be more or less clear. The brightness of an object depends on the way the object (surface) is lighted, that is, how more light, how intenser/brighter the colour. Only colours perceived to belong to an area seen in isolation from other colours exhibit brightness. When we consider a colour as a related colour (a colour perceived to belong to an area seen in relation to other colours), the observed brightness of a point of the coloured object depends then not only on the light intensity in that point but also on the light intensity in other points of the area or background. So we define the term **lightness** as the relative brightness of an area compared to the brightness of an equally illuminated white surface. Brightness refers to the absolute perception of the amount of light of the colour element of interest, while lightness can be seen as the relative brightness. Only related colours exhibit lightness.

$$\text{Lightness} = \frac{\text{Brightness}}{\text{Brightness(white)}}$$

The lightness of the used background can cause a difference in the lightness of the object. This we call the **lightness contrast effect**, and is illustrated in figure 3.9.
Each small grey square has the same physical brightness (intensity), that is, our eyes receive exactly the same amount of light from each of them. So we expect that they all would be the same value of grey, that our eyes observe them all as the same grey

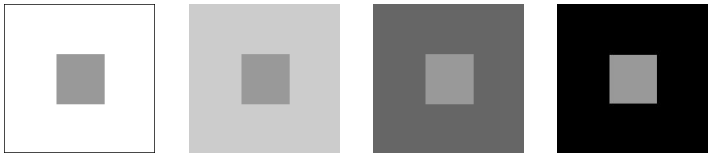


Figure 3.9: The lightness contrast effect.

value. But the grey square at the dark grey background seems to be much brighter than the grey square at the light grey background: the darker the background, the lighter the little grey square appears.

3.2.3 YUV, YIQ and YCbCr Colour Model

These colour models are called the television transmission colour models. The YUV and YIQ model are used in colour television broadcasting (YUV in Europe and YIQ in the USA). These colour models are a recoding of RGB that are used for the coding of colour images in TV and in colour video, and are also very useful for transmitting television signals for black-white TV. The YCbCr colour model is independent of coding systems for TV signals and is used for the representation of TV images in digital systems.

These colour models are based on the fact that the human eye is more sensitive for changes in brightness than for changes in hue or saturation. The Y-component is identical to the Y-component in the CIE XYZ colour model (see section 3.2.5) and contains information about the brightness of an image, while the other components encode information about the colour, that is, hue and saturation. Because the eye is most sensitive for changes in brightness, these models use a larger range for the Y-component than for the U- or V-component (I- or Q-component and Cb- or Cr-component): the data rate can be shared as follows: Y:U:V 4:2:2 or Y:U:V 4:1:1. The Y-component contains all information needed for black-white television, that is, the Y-component gives the greyscale image of a colour image. Figure 3.10 illustrates this.

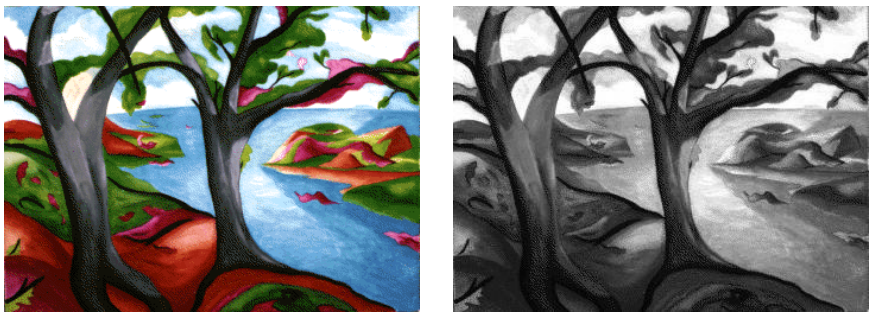


Figure 3.10: The original colour image (left) and the greyscale image expressed in the Y-component (right).

3.2.4 HSV and HSL Colour Model

These two colour models use equivalent axes in their representation of colour. In the HSV colour model a colour is defined by the three components hue, saturation and value and in the HSL (also called HSI) colour model the three quantities hue, saturation and luminance (= intensity) are used to characterise a colour.

We have seen (opponent process theory) that all colour hues can be arranged in an

opponent colour wheel along two axes (red-green and blue-yellow) so that all colours along the red-green axis contain some red on one side of the wheel, while all the colours on the other side contain some green, and so that all colours along the blue-yellow axis have some blue on one half of the wheel, while the colours that remain in the opposite half of the wheel appear yellowish. This way we can range the hue component from 0 to 2π, which usually begins and ends by red. Saturation indicates the colour purity (lack of white in the colour). Values for the saturation component range from 0% if the colour is not saturated (grey values) to 100% if the colour is completely saturated (pure colours). The intensity component of a colour in the HSL colour model gives the intensity (brightness/lightness) of that colour and varies from 0 to 1, where the corresponding colours become increasingly brighter. Maximum intensity is sensed as pure white, minimum intensity as pure black. The other colour model HSV differs in the formulae for the values of the intensity. Fully saturated colours with different hues have the same value $V = 1$ in HSV and the same lightness $L = 0.5$ in HSL. Nevertheless, this is not always true in human perception, e.g., fully saturated yellow is always lighter than fully saturated blue. In the HSL colour model we find the brightest, most intense colour at a lightness value of exactly half of the maximum. Colours with a lightness percentage smaller than half of the maximum lightness value are darker while colours with a percentage larger than the maximum are lighter. In figure 3.11 and 3.12 [87] we illustrate a graphical representation of the HSV and HSL colour model respectively. Both the HSV and HSL colour model are used in computer graphics applications.

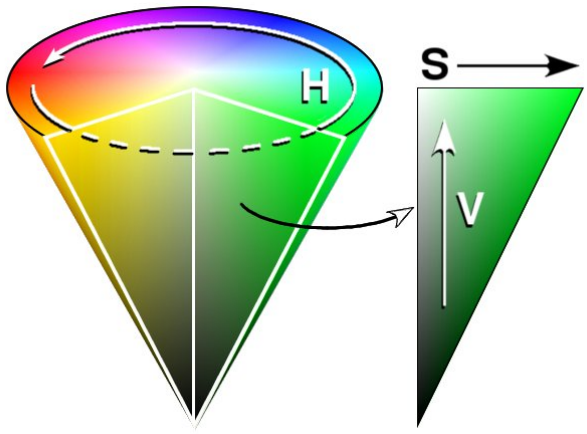


Figure 3.11: A graphical representation of the HSV colour model.

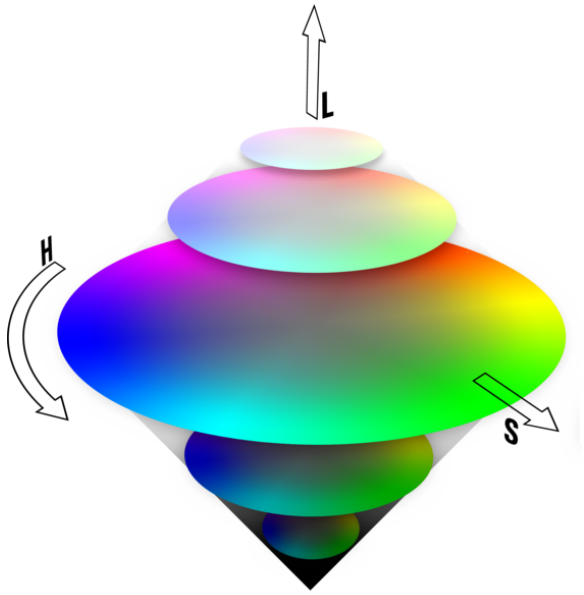


Figure 3.12: A graphical representation of the HSL colour model.

3.2.5 CIE Colour Models

All colour models we have seen until now express colour in a device dependent way. The additive colour models we have described, e.g. the RGB, YUV and YIQ colour model, are a.o. used for television or computer screens. Additive colours depend on the kind of screen that is used. Colours in the subtractive colour models we mentioned, e.g. CMYK colours, vary with printer, ink and paper characteristics.
That is why the CIE (Commission Internationale de l'Eclairage) has created a number of (both additive and subtractive) colour models that are device independent. Colours can be specified in the CIE-based colour models in a way that is independent of the characteristics of any particular display or reproduction device. The CIE colour specification system provides a standard method for specifying a colour under controlled viewing conditions, that is, the CIE system standardises three key elements of colour perception: light, object and eye. We would not go here into more detail, but the interested reader can find more information in [58].

CIE XYZ colour model

The CIE XYZ colour model is special because it is based on direct measurements of the human eye. The CIE has done experiments to determine how primary colours should

be mixed to reproduce colours. The experiments showed that to produce some colours, a component of light has to be subtracted. When we add two primaries together, the saturation decreases. For this reason it can be impossible to produce a maximal saturated colour by adding two colours together. For example, if we use the CIE RGB primaries at 700, 546.1 and $435.8\ nm$ respectively, and we want to obtain a colour with a wavelength of about $500\ nm$, we see in figure 3.13 [87] that blue and green is needed. But adding blue and green together will produce a desaturated $500\ nm$ colour so that red has to be added to the colour we want to produce, to saturate it. If $S\ 500\ nm$ represents the saturated $500\ nm$ colour we want to obtain, then we get

$$\begin{aligned} S\ 500\ nm + R &= G + B \\ &or \\ S\ 500\ nm &= G + B - R. \end{aligned}$$

The negative value means that the spectral colour red has to be subtracted.
It is interesting to note that because TV and computer screens use additive mixtures of red, green and blue light, they cannot produce colours with wavelengths around $500\ nm_{CIERGB}$.

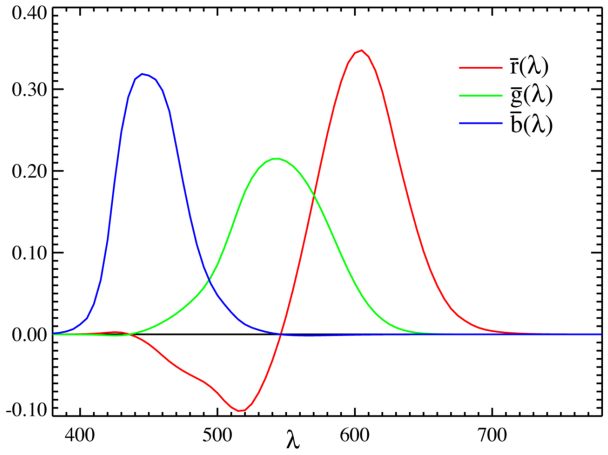


Figure 3.13: The colour matching functions for the CIE RGB primaries.

To avoid the problem of having negative components, the CIE has introduced three supersaturated (not physically reproducible) **tristimulus values** X, Y and Z, which quantify a colour by defining the amounts of the CIE R, G and B primaries required to match the colour by a standard observer under a particular light source so that additively mixing of the three 'imaginary' colours X, Y and Z can produce all human perceivable colours. These positive X, Y, Z values form the first CIE colour model, the CIE XYZ colour model. But this colour model is difficult to use because X, Y and Z separately

do not correspond to real colours. The X, Y, Z values are called the **standard colour coordinates**. All CIE based colour models are deduced of the XYZ model. The Y tristimulus value is a parameter of the brightness of a colour. Figure 3.14 [87] shows the colour matching functions for the CIE XYZ tristimulus values.

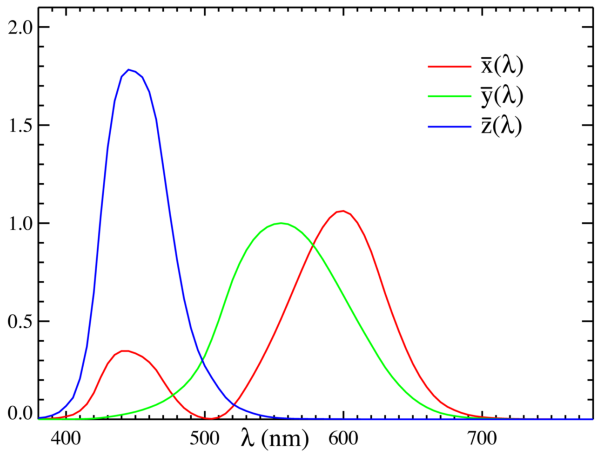


Figure 3.14: The colour matching functions for the CIE XYZ tristimulus values.

CIE Yxy colour model

Since the human eye has three types of colour sensors, a full plot of all visible colours is a three-dimensional figure. However, the concept of colour can be divided into two parts: intensity and chrominance (= hue and saturation), the part that carries 'colour'. Chrominance requires intensity to make it visible. For example, the colour white is a bright colour, while the colour grey is considered to be a less bright version of that same white. In other words, the chrominance of white and grey are the same, while their brightness differs. A colour does not change if the intensity changes. CIE has defined a colour model for representing colours (into two dimensions) independent of the 'intensity' of the colour. A useful two-dimensional representation of colours is obtained when the tristimulus values are normalized to lie in the unit plane, the plane over which the tristimulus values sum up to unity. The coordinates of the normalized tristimulus vectors are called **chromaticity coordinates** and a plot of colours on the unit plane using these coordinates is called a **chromaticity diagram**. The most commonly used chromaticity diagram is the CIE xy chromaticity diagram. The CIE xyz chromaticity coordinates can be obtained from the X, Y, Z tristimulus values in the

CIE XYZ colour model as follows

$$\begin{aligned} x &= X/(X+Y+Z) \\ y &= Y/(X+Y+Z) \\ z &= Z/(X+Y+Z). \end{aligned}$$

Only two of these three coordinates are independent; one use x and y at most. Thus, a colour (independent of intensity) can be specified by its coordinates (x, y). A colour, dependent on intensity, can then be specified by the triple (x, y, Y). Colours of objects (or printing inks) of course depend on the intensity (of the light source) as well.

Figure 3.15 [87] shows the CIE xy chromaticity diagram, where colours are points, obtained from the CIE XYZ colour model with D65 as used illuminant (see L*a*b* and L*u*v* colour model). A property of the chromaticity diagram is that a point that is an additive mixture of two colours always lies on the connection line of the chromaticity points of these two colours. This property is important for the measure of colour in colour displays, TV and the lightning industry.

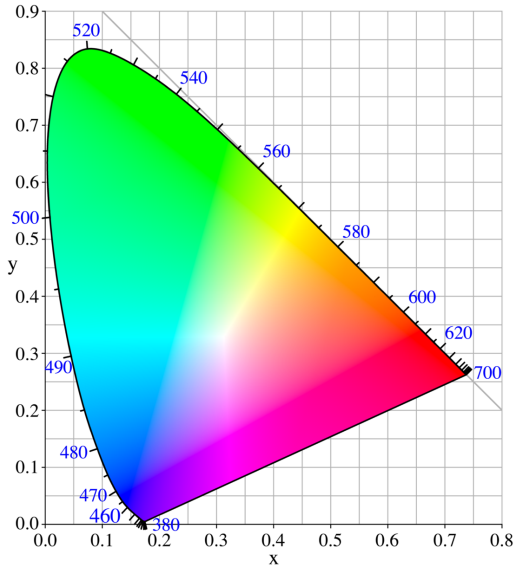


Figure 3.15: The xy chromaticity diagram.

Now, it is desirable if a distance on a chromaticity diagram corresponds to the degree of difference between two colours. The idea of measuring the difference between two colours was developed by D.L. MacAdam and summarized in the concept of a

MacAdam ellipse [40]. A MacAdam ellipse is the region on a chromaticity diagram that contains all colours that are indistinguishable to the average human eye, from the colour at the centre of the ellipse. As such it defines the concept of distance. Each of the ellipses are, by definition, circles of equal radius, and the only reason that they appear to be ellipses of different sizes in the CIE xy chromaticity diagram (see figure 3.16 [87]) is because the CIE xy space is warped (with respect to this metric). Based on the work of MacAdam, the CIE provided two perceptually uniform colour models for practical applications: the CIE L*u*v* and CIE L*a*b* colour model, both of which were designed to have an equal distance in the colour model corresponding to equal differences in colour, as measured by MacAdam.

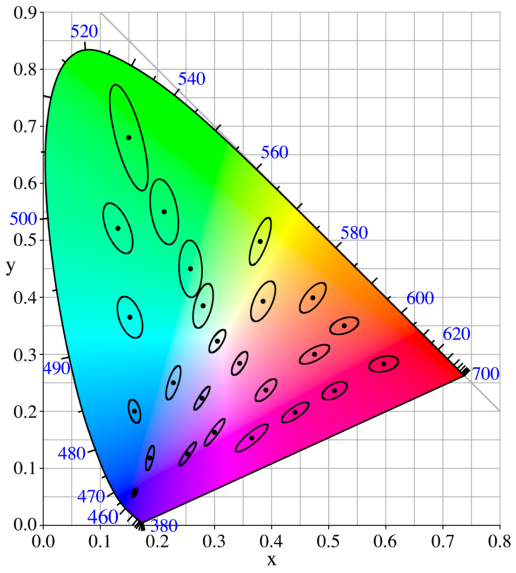


Figure 3.16: MacAdam ellipses plotted on the xy chromaticity diagram.

L*a*b* (CIELAB) and L*u*v* (CIELUV) colour model

There is one problem with the representation of colours in the XYZ and Yxy colour model: the presentation is not linear; colours that are close to each other in the XYZ or Yxy space can be perceived very differently for the human eye and colours that seem to be the same for the human eye can lie far from each other in these colour models. The L*a*b* and the L*u*v* colour model are transformations of the XYZ colour model designed to obtain a linear colour model, that is, to have a uniform correspondence between geometric distances and perceptual distances between colours that are seen under the same light source; equal Euclidean distances correspond to roughly equal

perceived colour differences. We get uniform colour models where small colour differences (changes) can be quantified by the Just Noticeable Colour Difference or JND (the distance between two almost indistinguishable colours). The most important and common light source is daylight, which is represented as illuminant D65 and D50 for the surface colour industries (textiles, paint and plastics) and the graphic arts industry respectively. Artificial light sources are also widely used such as white fluorescent light (illuminant F) and incandescent light (illuminant A). Both colour models L*a*b* and L*u*v* represent colours relative to a reference white point, which is defined as the whitest light that can be generated by a given device represented in terms of the XYZ colour model. Because the L*u*v* and L*a*b* colour model represent colours relative to this definition of white light, they are not completely device independent: two numerically equal colours are truly identical only if they are measured relative to the same white point.

Figure 3.17 [75] gives a graphical representation of the L*a*b* colour model.

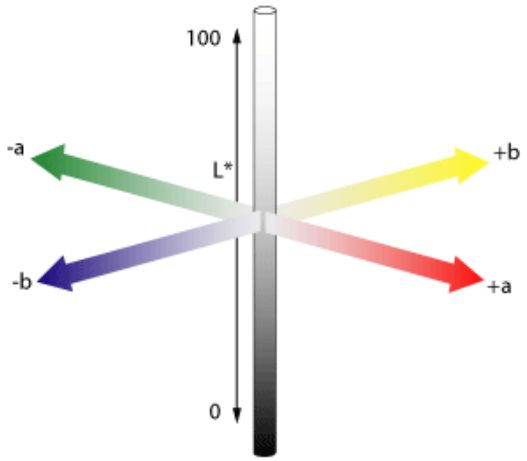


Figure 3.17: Graphical representation of the L*a*b* colour model.

Both colour models L*a*b* and L*u*v* use a common lightness scale L*, which depends only on the value Y. The vertical axis L* in the centre of both colour models represents the lightness where the values for the lightness range from 0 (black) to 100 (white). Both colour models use different uniform colour axes: the colour axes a* versus b* and u* versus v* (red-green versus yellow-blue) are based on the fact that a colour cannot be red and green at the same time or both blue and yellow because these colours are opposite (opposite colour theory). At every colour axis values go from positive to negative. At the a* and u* axis the positive values give the amount of red and the negative values the amount of green, while at the b* and v* axis yellow is

positive and blue negative. For these axes 0 is neutral grey.

L*a*b* is a colour model that is used to represent subtractive colour systems, and is therefore used in different colour imaging and printing industries like in the textile, while L*u*v* is a colour model that is used to represent additive colour systems and is very useful in applications with additive mixture of light in the display industry like in videos and PCs.

More information about the reception and reproduction of colour can be found in [19], [55], [58], [72] and [73]. For colour blindness we refer to [17] and [87]. An extensive release about digital colour imaging can be found in [73].

Chapter 4

Colour Morphology

In the first and second section of this chapter we repeat the definition of the basic operators dilation, erosion, closing and opening for binary and greyscale morphology. For a detailed study of binary and greyscale morphology we refer to [9, 10, 11, 25, 26, 44, 57]. Thereafter, a state-of-the-art literature study of colour morphology shows that there already exist some nice extensions of MM to colour. To apply morphological operators to colour images we need the concept of a supremum and infimum, and thus, of an ordering in the used colour model. After having described the 'trivial' approach of processing the morphological operators on each of the colour components separately, we will present in this work a new vector-based approach for the extension of MM for greyscale images to colour morphology. We will extend the basic morphological operators dilation and erosion based on the threshold, umbra and fuzzy set approach to colour images. Colour images can be modelled using different colour models; here our approach is described in the RGB [15, 16], HSV [15] and L*a*b* [15] colour model. We look at colours as a whole, namely as vectors, and order these colour vectors in each of the three colour models, and so colour morphological operators are presented accordingly. The colour models RGB, HSV and L*a*b* become with this new ordering and associated minimum and maximum operators a complete lattice on which we have defined a negator, some t-norms and implicators. Experimental results show that our method provides an improvement on the component-based approach of morphological operators applied to colour images and achieves similar or better results than those obtained by other methods.

4.1 Binary Morphology

Consider a binary image A and a binary **structuring element** B, which is also an image but very small in comparison with A and has to be chosen by the morphologist.

The **translation** $T_y(B)$ **of** B **by a vector** $y \in \mathbb{R}^2$ is defined as

$$T_y(B) = \{x \in \mathbb{R}^2 \mid x - y \in B\};$$

the **reflection of** B is defined as $-B = \{-x \in \mathbb{R}^2 \mid x \in B\}$.

Definition 4.1. *Let A be a binary image and B a binary structuring element. The **binary dilation** $D(A, B)$ **of** A **by** B is the binary image given by*

$$D(A, B) = \{y \in \mathbb{R}^2 \mid T_y(B) \cap A \neq \emptyset\}.$$

*The **binary erosion** $E(A, B)$ **of** A **by** B is defined as*

$$E(A, B) = \{y \in \mathbb{R}^2 \mid T_y(B) \subseteq A\}.$$

*If A is a binary image and B is a binary structuring element, the **binary closing** $C(A, B)$ and the **binary opening** $O(A, B)$ **of** A **by** B are the binary images*

$$\begin{aligned} C(A, B) &= E(D(A, B), -B) \\ O(A, B) &= D(E(A, B), -B). \end{aligned}$$

Equivalent expressions for $D(A, B)$ and $E(A, B)$ are

$$\begin{aligned} D(A, B) &= \{y \in \mathbb{R}^2 \mid (\exists b \in B)(y + b \in A)\} \\ &= \bigcup_{b \in B} T_{-b}(A), \\ E(A, B) &= \{y \in \mathbb{R}^2 \mid (\forall b \in B)(y + b \in A)\} \\ &= \bigcap_{b \in B} T_{-b}(A). \end{aligned}$$

For the closing $C(A, B)$ and opening $O(A, B)$ we can also write

$$\begin{aligned} C(A, B) &= \{y \in \mathbb{R}^2 \mid (\forall z \in \mathbb{R}^2)(y \in T_z(B) \Rightarrow T_z(B) \cap A \neq \emptyset)\} \\ O(A, B) &= \{y \in \mathbb{R}^2 \mid (\exists z \in \mathbb{R}^2)(y \in T_z(B) \wedge T_z(B) \subseteq A)\}. \end{aligned}$$

The binary dilation and erosion have a beautiful geometrical interpretation, see figure 4.1. The dilation $D(A, B)$ contains all points y in $\mathbb{R}^2$ for which the translation $T_y(B)$ of the structuring element B has a non-empty intersection with the image A. A point y belongs to the dilation $D(A, B)$ if and only if the translation $T_y(B)$ and A hit each other. The binary erosion $E(A, B)$ consists of all points $y \in \mathbb{R}^2$ for which the translation $T_y(B)$ of B is contained in A. A point y belongs to the erosion $E(A, B)$ if and only if the translation $T_y(B)$ and $co(A)$ do not hit. For an example we refer to figure

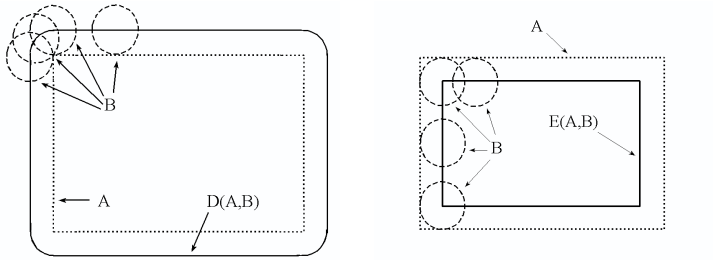


Figure 4.1: Geometrical interpretation of the binary dilation (left) and the binary erosion (right). The centre of the structuring element B coincides with the origin of the coordinate system.

4.2. The closing $C(A, B)$ consists of all points $y \in \mathbb{R}^2$ for which any translation of B that contains y has a non-empty intersection with A. The opening $O(A, B)$ consists of all points $y \in \mathbb{R}^2$ for which any translation of B that contains y is contained in A. When we use the following structuring element B given by (the underlined element corresponds to the origin of coordinates)

$$B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \underline{1} & 1 \\ 1 & 1 & 1 \end{pmatrix},$$

we get the following results for the binary dilation and erosion

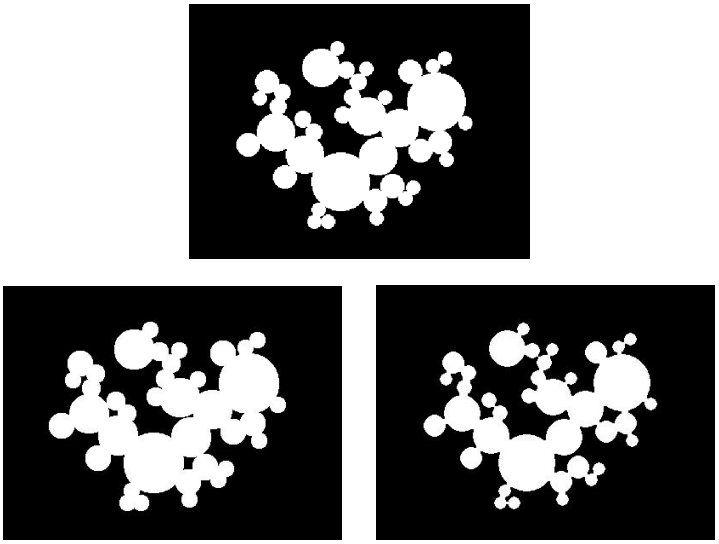


Figure 4.2: At the top: the original binary image A, at the bottom: the binary dilation $D(A, B)$ (left) and the binary erosion $E(A, B)$ (right). You see that the dilation enlarges the objects in the image, while the erosion reduces them.

Property 4.2. *[43] Let A be a binary image and B a binary structuring element, then it holds that*

$$E(A, B) \subseteq D(A, B) \text{ and } O(A, B) \subseteq C(A, B).$$

If B contains the origin (i.e., $\mathbf{0} \in B$), then is

$$E(A, B) \subseteq O(A, B) \subseteq A \subseteq C(A, B) \subseteq D(A, B).$$

With this property the binary image $D(A, B) - E(A, B)$ can serve as an edge-image of the original image A, which we call the morphological gradient $G^B(A)$ of A. Analogously we have defined the external morphological gradient $G_e^B(A)$ and the internal morphological gradient $G_i^B(A)$ of A, which will give us the extern and inner edge-image of A respectively. Figure 4.3 illustrates this.

Application 4.3. *Let A be a binary image and B a binary structuring element. The **morphological gradient** G^B is defined as*

$$G^B(A) = D(A, B) \backslash E(A, B).$$

*If B contains the origin, we define the **extern morphological gradient** G_e^B as*

$$G_e^B(A) = D(A, B) \backslash A,$$

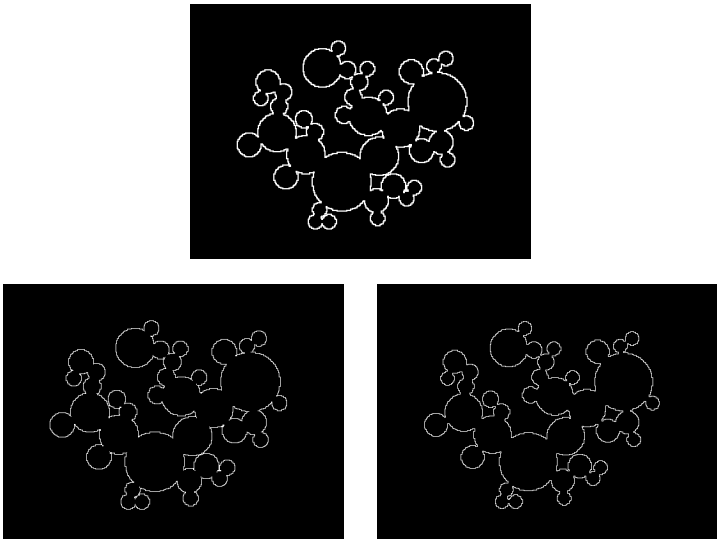


Figure 4.3: At the top: the morphological gradient $G^B(A)$ of A, at the bottom: the extern morphological gradient $G_e^B(A)$ (left) and the intern morphological gradient $G_i^B(A)$ (right) of A.

*and the **intern morphological gradient** G_i^B as*

$$G_i^B(A) = A \backslash E(A, B).$$

4.2 Greyscale Morphology

4.2.1 Greyscale Morphology Based on the Threshold Approach

Consider a greyscale image A represented as a $\mathbb{R}^2 - [0, 1]$ mapping and a binary structuring element B modelled as a crisp subset of $\mathbb{R}^2$.
The **support** d_A **of** A is defined as the set $d_A = \{x \in \mathbb{R}^2 \mid A(x) > 0\}$;
the **reflection of** A is the $\mathbb{R}^2 - [0, 1]$ mapping $-A$ characterised by $(-A)(x) = A(-x)$, for all x in $\mathbb{R}^2$.

Definition 4.4. *Let A be a greyscale image and B a binary structuring element. The **t-dilation** $D_t(A, B)$ and the **t-erosion** $E_t(A, B)$ are the greyscale images given by*

$$D_t(A, B)(y) = \sup_{x \in T_y(B)} A(x) \quad \textit{for } y \in \mathbb{R}^2,$$

$$E_t(A, B)(y) = \inf_{x \in T_y(B)} A(x) \quad \textit{for } y \in \mathbb{R}^2.$$

*The **t-closing** $C_t(A,B)$ and the **t-opening** $O_t(A,B)$ are then defined as*

$$\begin{aligned} C_t(A,B) &= E_t(D_t(A,B),-B), \\ O_t(A,B) &= D_t(E_t(A,B),-B). \end{aligned}$$

For the t-closing and t-opening we can also write

$$\begin{aligned} C_t(A,B) &= \inf_{z\in T_y(-B)} \Big(\sup_{x\in T_z(B)} A(x) \Big), \\ O_t(A,B) &= \sup_{z\in T_y(-B)} \Big(\inf_{x\in T_z(B)} A(x) \Big). \end{aligned}$$

Property 4.5. *[12] Property 4.2 also applies to greyscale morphology based on the threshold approach.*

Application 4.6. *The **t-morphological gradient** G_t^B, **extern t-morphological gradient** $G_{t,e}^B$ and **intern t-morphological gradient** $G_{t,i}^B$ can be defined analogously as in the case of binary morphology.*

Figure 4.4 gives an example of the t-morphological operators dilation and erosion, and the t-morphological gradient.

Figure 4.4: At the top: the original greyscale image A (left) and the t-morphological gradient $G_t^B(A)$ (right), at the bottom: the t-dilation $D_t(A, B)$ (left) and the t-erosion $E_t(A, B)$ (right).

4.2.2 Greyscale Morphology Based on the Umbra Approach

Let A be a greyscale image and B a greyscale structuring element, both modelled as $\mathbb{R}^2 - \overline{\mathbb{R}}$ mappings. In practice we can restrict to $\mathbb{R}^2 - [0,1]$ mappings because the interval $[0,1]$ corresponds to the universe of grey values, but the theory considers $\mathbb{R}^2 - \overline{\mathbb{R}}$ mappings. The **support of** A is defined as $d_A = \{x \in \mathbb{R}^2 \mid A(x) > -\infty\}$.

Definition 4.7. *Let A be a greyscale image and B a greyscale structuring element. The **u-dilation** $D_u(A,B)$ and the **u-erosion** $E_u(A,B)$ are the greyscale images defined as*

$$D_u(A,B)(y) = \sup_{x \in T_y(d_B)} A(x) + B(x-y) \quad \textit{for } y \in \mathbb{R}^2,$$

$$E_u(A,B)(y) = \inf_{x \in T_y(d_B)} A(x) - B(x-y) \qquad \textit{for } y \in \mathbb{R}^2.$$

*Let A be a greyscale image and B a greyscale structuring element, the **u-closing** $C_u(A,B)$ and the **u-opening** $O_u(A,B)$ are the greyscale images given by*

$$C_u(A,B) = E_u(D_u(A,B),-B),$$
$$O_u(A,B) = D_u(E_u(A,B),-B).$$

Explicit expressions for the u-closing and u-opening are, for all $y \in \mathbb{R}^2$,

$$C_u(A,B)(y) = \inf_{z \in T_y(-d_B)} (\sup_{x \in T_z(d_B)} (\Big(A(x) + B(x-z) \Big) - B(y-z)))$$

$$O_u(A,B)(y) = \sup_{z \in T_y(-d_B)} (\sup_{x \in T_z(d_B)} (\Big(A(x) - B(x-z) \Big) + B(y-z))).$$

Property 4.8. *[43] Property 4.2 for binary images holds on for greyscale morphology based on the umbra approach. The condition $\mathbf{0} \in B$ has to be replaced by $B(\mathbf{0}) \geq 0$.*

Application 4.9. *Again, the definition of the **u-morphological gradient** G_u^B, **extern u-morphological gradient** $G_{u,e}^B$ and **intern u-morphological gradient** $G_{u,i}^B$ are similar to those for greyscale morphology based on the threshold approach.*

4.2.3 Fuzzy Mathematical Morphology

Since greyscale images can be modelled as $\mathbb{R}^2 - [0,1]$ mappings, we can identify greyscale images with fuzzy sets and extend binary morphology to greyscale morphology using fuzzy set theory.

Definition 4.10. *Let A be a greyscale image and B a greyscale structuring element (both seen as fuzzy sets), $\mathcal{C}$ a conjunctor on $[0,1]$ and $\mathcal{I}$ an implicator on $[0,1]$. The **fuzzy dilation** $D_{\mathcal{C}}(A,B)$ and the **fuzzy erosion** $E_{\mathcal{I}}(A,B)$ are the fuzzy sets defined as*

$$D_{\mathcal{C}}(A,B)(y) = \sup_{x \in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) \quad \textit{for } y \in \mathbb{R}^2,$$

$$E_{\mathcal{I}}(A,B)(y) = \inf_{x \in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) \qquad \textit{for } y \in \mathbb{R}^2.$$

*The **fuzzy closing** $C_{\mathcal{C},\mathcal{I}}(A,B)$ and the **fuzzy opening** $O_{\mathcal{C},\mathcal{I}}(A,B)$ are the fuzzy sets given by*

$$\begin{aligned} C_{\mathcal{C},\mathcal{I}}(A,B) &= E_{\mathcal{I}}(D_{\mathcal{C}}(A,B),-B), \\ O_{\mathcal{C},\mathcal{I}}(A,B) &= D_{\mathcal{C}}(E_{\mathcal{I}}(A,B),-B). \end{aligned}$$

Property 4.11. *[43] Because $\mathcal{T}_M \geq \mathcal{T}_P \geq \mathcal{T}_W$ and $\mathcal{I}_{KD} \leq \mathcal{I}_R \leq \mathcal{I}_W$, we obtain*

$$D_{\mathcal{T}_M}(A,B) \supseteq D_{\mathcal{T}_P}(A,B) \supseteq D_{\mathcal{T}_W}(A,B)$$

and

$$E_{\mathcal{I}_{KD}}(A,B) \subseteq E_{\mathcal{I}_R}(A,B) \subseteq E_{\mathcal{I}_W}(A,B),$$

for every greyscale image A and greyscale structuring element B.

Application 4.12. *Edge detection can be done in the same way as before.*

Figure 4.5 and figure 4.6 illustrate the fuzzy dilation and fuzzy erosion using the following greyscale structuring element

$$B(i,j,1) = B(i,j,2) = B(i,j,3) = \frac{1}{255}\begin{pmatrix} 200 & 220 & 200 \\ 220 & \underline{255} & 220 \\ 200 & 220 & 200 \end{pmatrix},\ 1 \leq i,j, \leq 3.$$

Figure 4.5: At the top: the original greyscale image A (left) and the fuzzy dilation $D_{\mathcal{T}_M}(A, B)$ (right), at the bottom: the fuzzy dilation $D_{\mathcal{T}_P}(A, B)$ (left) and the fuzzy dilation $D_{\mathcal{T}_W}(A, B)$ (right).

Figure 4.6: At the top: the original greyscale image A (left) and the fuzzy erosion $E_{\mathcal{I}_{KD}}(A, B)$ (right), at the bottom: the fuzzy erosion $E_{\mathcal{I}_R}(A, B)$ (left) and the fuzzy erosion $E_{\mathcal{I}_W}(A, B)$ (right).

4.3 Colour Morphology

So far we have studied greyscale morphology based on the threshold approach, on the umbra approach and on fuzzy set theory, on the unit interval $[0,1]$. Notice that $[0,1]$ with the ordinary ordering is a lattice, even more, it is a complete lattice.

Colour images can be represented (possibly after scaling) as $\mathbb{R}^2 - [0,1] \times [0,1] \times [0,1]$ mappings. A first way to extend mathematical morphology for greyscale images to colour images is the component-based approach. Mathematical morphology can be naturally extended to colour morphology by processing the morphological operators on each of the colour components separately, where we get again a complete lattice with the product ordering.

- Given the complete lattices $(\mathcal{L}_1, \leq_1), (\mathcal{L}_2, \leq_2), \ldots, (\mathcal{L}_d, \leq_d)$. Define $\mathcal{L} = \mathcal{L}_1 \times \mathcal{L}_2 \times \ldots \times \mathcal{L}_d$, that is, $\mathcal{L}$ contains all d-tuples $(x_1, x_2, \ldots, x_d)$ with $x_k \in \mathcal{L}_k$ for $k = 1, 2, \ldots, d$. Define the relation $\leq$ on $\mathcal{L}$, for all $(x_1, x_2, \ldots, x_d)$ and $(y_1, y_2, \ldots, y_d)$ in $\mathcal{L}$, by

  $$(x_1, x_2, \ldots, x_d) \leq (y_1, y_2, \ldots, y_d) \text{ iff } x_k \leq_k y_k, \ \forall k = 1, \ldots, d.$$

  We call this ordering the product ordering. $(\mathcal{L}, \leq)$ is a complete lattice.

- Consider a t-norm $\mathcal{T}_1$ on a complete lattice $(\mathcal{L}_1, \leq_1)$ and a t-norm $\mathcal{T}_2$ on a complete lattice $(\mathcal{L}_2, \leq_2)$. The direct product $\mathcal{T}_1 \times \mathcal{T}_2$ of $\mathcal{T}_1$ and $\mathcal{T}_2$ defined, for all (x_1, x_2) and (y_1, y_2) in $\mathcal{L}_1 \times \mathcal{L}_2$, as

  $$\mathcal{T}_1 \times \mathcal{T}_2((x_1, x_2), (y_1, y_2)) = (\mathcal{T}_1(x_1, y_1), \mathcal{T}_2(x_2, y_2))$$

  is a t-norm on the product lattice $(\mathcal{L}', \leq) = (\mathcal{L}_1 \times \mathcal{L}_2, \leq)$.

- Let $\mathcal{I}_1$ be an implicator on a complete lattice $(\mathcal{L}_1, \leq_1)$ and $\mathcal{I}_2$ an implicator on a complete lattice $(\mathcal{L}_2, \leq_2)$. The direct product $\mathcal{I}_1 \times \mathcal{I}_2$ of $\mathcal{I}_1$ and $\mathcal{I}_2$ defined, for all (x_1, x_2) and (y_1, y_2) in $\mathcal{L}_1 \times \mathcal{L}_2$, as

  $$\mathcal{I}_1 \times \mathcal{I}_2((x_1, x_2), (y_1, y_2)) = (\mathcal{I}_1(x_1, y_1), \mathcal{I}_2(x_2, y_2))$$

  is an implicator on the product lattice $(\mathcal{L}', \leq) = (\mathcal{L}_1 \times \mathcal{L}_2, \leq)$.

A major disadvantage of this approach is that the existing correlations between the different colour components are not taken into account and this often leads to disturbing artefacts. Another approach is to treat the colour at each pixel as a vector. Because we need the concept of a supremum and infimum to define morphological operators, we first have to define an ordering between these colour vectors. We have considered the three most common used colour models RGB, HSV and L*a*b*.

A colour in the **RGB** colour model is obtained by adding the three colours red, green

and blue in different combinations. Therefore a colour can be defined as a vector in a three-dimensional space that can be represented as a unit cube using a Cartesian coordinate scheme. This way every point in the cube represents a vector (colour). The greyscale spectrum is characterised by the line between the black top Bl with coordinates $(0, 0, 0)$ and the white top $\mathrm{Wh}(1, 1, 1)$.

In the **HSV** colour model a colour is characterised by the three quantities hue, saturation and value. Because of the opposite colour theory all colour hues can be arranged in an opponent colour wheel along two axes (red-green and blue-yellow) that begins and ends by the same colour. So we can range the hue component in a circle from 0 to 2π, which usually begins and ends by red. Values for the saturation component range from 0 if the colour is not saturated (grey values) to 1 if the colour is completely saturated (pure colours). The value component in the HSV colour model varies from 0 (black) to 1 (white), where the colours become increasingly brighter. While adding black to a certain colour, the value of the colour will decrease. The value axis begins by black, ends by white and in between we get all shades of grey. The three-dimensional colour model HSV is usually represented as a cone.

The colour models **L*a*b*** and **L*u*v*** use a common lightness scale L*. The vertical axis L* in the centre of both colour models represents the lightness/brightness of a colour where the values range from 0 (black) to 1 (white), with in between grey values. Both colour models use different uniform colour axes: the colour axes a* versus b* and u* versus v* (red-green versus yellow-blue) are based on the fact that a colour cannot be red and green at the same time or both blue and yellow because these colours are opposite (opposite colour theory). At every colour axis values range from positive to negative. At the a* and u* axis the positive values give the amount of red and the negative values the amount of green, while at the b* and v* axis yellow is positive and blue negative. For these axes 0 is neutral grey.

Remark that in practice only a finite number of colours can be obtained in a colour model. Since each colour component is usually stored as 8 bits (one byte), i.e., the values of each colour component range in the interval $\left[0, 2^8 - 1\right]$, a colour in a three-dimensional colour model is stored as a 24-**bit colour**. The values of each colour component by storage usually range in the interval $[0, 255]$, but we can always scale them to the interval $[0, 1]$. So we always work with finite colour models.

4.3.1 State-of-the-Art Overview of Colour Morphology

Because there is no unambiguous way to order two or more colours, there exist different techniques to order colour. The most common used techniques are the **component-wise ordering**, also called **marginal ordering**, **reduced ordering**, **partial ordering** and **conditional ordering**, also called **lexicographical ordering**.

In **marginal ordering** each colour component is ordered independently and the operations are applied to each colour component of the image. But this approach does not exploit the correlation between the different colour components and because of this new colours can be introduced in an image.
In **reduced ordering** a single value is given to each multivariate value. So to each colour (vector) c_i in the considered colour model is a scalar value $d_i = d(c_i)$, normally $d : \mathbb{R}^3 \to \mathbb{R}$, added. After d_i has been obtained for each i, the vectors $c_1, \ldots, c_n$ are ordered based on $d_1, \ldots, d_n$ as follows:

$$c_{(1)} \leq \ldots \leq c_{(n)},$$

where $c_{(r)}$ is the vector corresponding with the scalar value $d_{(r)}$, the rth smallest element of the set $\{d_1, d_2, \ldots, d_n\}$. The output vector at each point in the image is, by definition of this ordering, one of the vectors in the original image so there is no possibility of introducing new colour vectors into the image. Usually some type of distance metric is used to perform reduced ordering. The output will of course depend on the used scalar-valued function, where characteristics of the human visual system, such as luminance, can be used as metric.
In **partial ordering** the colour vectors are partitioned into groups, which are then ordered.
In **conditional ordering** the colours are ordered using one component initially. In case multiple colours have the same initial component values, a second component is used to order the colours, and so on. Let $c = \{c_1, c_2, \ldots, c_n\}$ and $c' = \{c'_1, c'_2, \ldots, c'_n\}$ be two colour vectors in the considered colour model ($n \in \mathbb{N}$). An example of a lexicographical order can be

$$c < c' \quad \text{if} \quad \begin{cases} c_1 < c'_1 \quad \text{or} \\ c_1 = c'_1 \text{ and } c_2 < c'_2 \quad \text{or} \\ c_1 = c'_1 \text{ and } c_2 = c'_2 \ \ldots \ c_n < c'_n \end{cases} .$$

This way we get a total ordering. This approach makes sense when a priority can be placed on the components, but this is of course not the case when dealing with the RGB colour model.

In [56] the authors show that the generalisation of morphological operations to complete lattices is necessary for a mathematically coherent application of morphological operators to greyscale images. A computer implementation of mathematical morphology will work with digital images defined on a finite grid where the set of grey values is a bounded finite set of integers. A computation of grey values can thus give a value outside of this bounded set so that one gets an arithmetic overflow. The conclusion of their analysis is that the problem of grey value overflow can be dealt with correctly only by taking the complete lattice structure of the set of greyscale images into account.
In [77] the authors demonstrate that an artificial total ordering on multivariate data is the only way to use morphological operators on multivariate images while introducing

no new pixel values.

A number of possible orders for colour vectors in the RGB colour model have been proposed in [5, 8, 18, 33, 59]. In [5] an α-modulus lexicographical order in the RGB and HSL colour model is proposed, where the choice of the value for α controls the degree of influence of the first component with regard to the others, and makes the lexicographical order thus more flexible:

$$c <_\alpha c' \quad \text{if} \quad \begin{cases} (c_1/\alpha) < (c'_1/\alpha) \quad \text{or} \\ (c_1/\alpha) = (c'_1/\alpha) \text{ and } c_2 < c'_2 \quad \text{or} \\ (c_1/\alpha) = (c'_1/\alpha) \text{ and } c_2 = c'_2 \ \ldots \ c_n < c'_n \end{cases} .$$

Because in the RGB colour model no colour R, G or B plays a dominant role, the maximum and minimum of the three RGB values for every pixel are first calculated. The authors propose an α-modulus lexicographical order where the first component is given by, for every colour c in RGB,

$$I(c) = \beta \cdot (0.3 \cdot c_r + 0.6 \cdot c_g + 0.1 \cdot c_b) + (1-\beta) \cdot (\max(c_r, c_g, c_b) - \min(c_r, c_g, c_b)) .$$

The function I is a combination of the RGB components and the $\max - \min$ of the components, weighted by β. The other components for ordering are the red, then the green and finally the blue component. After deep test, the authors have found that the value $\beta = 0.8$ gives very good visual effects. This order is called $I - RGB_\alpha$. In [8] a reduced ordering in RGB is proposed. The used measurement functions are linear combinations of the tristimulus values, e.g. the luminance image $d(c) = 0.299 \cdot c_r + 0.587 \cdot c_g + 0.114 \cdot c_b$, and the Euclidean norm $d(c) = \sqrt{(c_r^2 + c_g^2 + c_b^2)}$. If two different colour values are ordered equally, the output can be chosen based on the position in the structuring element window, but no further analysis or example of this condition is made. New morphological operations are defined. The dilation selects that colour with the largest measure d_i and the erosion selects that colour with the smallest d_i. In [18] a new approach for the ordering of the RGB model is presented and applied to mathematical morphology, where the adaptation of a linear growing self-organizing network to the three-dimensional colour model allows the definition of an order relationship among colours. In [33] new colour morphological operators are defined after ordering the RGB colour vectors by using the first principal component analysis. On the basis of this reduced ordering, new infimum and supremum are determined. Using the new infimum and supremum, the fundamental erosion and dilation operators are defined. In [59] a new set of morphological operators for RGB colour images based on a combination of reduced and conditional ordering is proposed. The RGB colours are transformed into the $C - Y$ colour model, a colour television model. The distance to a reference colour vector, determined by its hue, provides the primary ordering criterion. The defined colour dilation will tend to move towards this reference colour, while the colour erosion will tend to move away from it. But the reference colour must have

maximum luminance and maximum saturation to obtain a total ordering of colour vectors, so we can speak of red dilation and blue erosion for example, but white or black cannot be used as reference colour vector. This is of course a disadvantage.

A number of possible orders [3, 35, 36, 37, 39] for colour vectors in the HSV and related colour models is proposed. In [3] the extension of morphological operators by using lexicographical orderings on the HSL system has been explored. A unified framework to consider different ways of defining morphological colour operators in a luminance, saturation and hue colour representation has been introduced. A new approach to colour mathematical morphology using a fuzzy model has been reported in [36]. It is based on a new vector ordering scheme in the HSV colour model that uses fuzzy if-then rules. The corresponding vector morphological operators of erosion and dilation are vector-preserving because no vector (colour) that is not present in the input data is generated and they produce unique results in every case. Moreover, the proposed operators possess the same basic properties like their greyscale counterparts. In [35] a new design and implementation of a fuzzy hardware structure for morphological colour image processing based on this new method has been proposed. In [37, 39] new partial colour vector orderings in the HSV colour model are presented. The new ordering is given, for two colours $c_1(h_1, s_1, v_1)$ and $c_2(h_2, s_2, v_2)$, by

$$c_1 < c_1 \quad \Leftrightarrow \quad \begin{cases} v_1 < v_2 \quad \text{or} \\ v_1 = v_2 \text{ and } s_1 > s_2 \end{cases} .$$

$$c_1 \leq c_1 \quad \Leftrightarrow \quad \begin{cases} v_1 < v_2 \quad \text{or} \\ v_1 = v_2 \text{ and } s_1 \geq s_2 \end{cases} .$$

$$c_1 = c_1 \quad \Leftrightarrow \quad \{ \ v_1 = v_2 \text{ and } s_1 = s_2 \quad .$$

The subtraction of two colours is defined as

$$c_1 - c_2 = c(h_1, s_1 - s_2, v_1 - v_2),$$

where $s_1 - s_2 = 0$ if $s_1 - s_2 < 0$ and $v_1 - v_2 = 0$ if $v_1 - v_2 < 0$.
The addition of two colours is defined as

$$c_1 + c_2 = c(h_1, s_1 + s_2, v_1 + v_2),$$

where $s_1 + s_2 = 1$ if $s_1 - s_2 > 1$ and $v_1 - v_2 = 1$ if $v_1 - v_2 > 1$. New infimum and supremum operators are defined, and so corresponding vector morphological u-operators, which are hue preserving.

In image analysis one often has to treat data distributed on the unit circle. Because hue is angle-valued and has no order of importance and no dominant position, it cannot be

ordered trivially. Some methods to apply mathematical morphology to data on the unit circle have been proposed. In [24] three possible approaches to apply morphological operators to circular data is presented. In the first approach a local origin, obviously variable at each image point, is chosen. The second approach considers grouping of circular data. A simple criterion to group data is introduced and basic morphological operators are defined so that they act only if a structuring element contains grouped data. The third approach defines a labelling on the unit circle to index the angles. If every pixel in the image is assigned a label, then we have an indexed partition of the image. If during the analysis of a set of data or an image, one has to choose an arbitrary origin before applying an operator, then one of these approaches can be used. In [22] morphological operators for the HSL colour model are presented. Some lexicographical vector orders are suggested. The first two orders use the two components luminance and saturation in the first position. The third order is a new saturation-weighted hue order, which uses the hue component in the first position, where the saturation values are used to weight the hue values. Paper [52] contains definitions for erosion and dilation for angle-valued images. The fundamental idea is to define a structuring element with a given hue or hues. From each image neighbourhood of the structuring element, the erosion returns the hue value that is closest to the hue of the corresponding structuring element member and the dilation returns the hue value that is farthest to the hue of the corresponding structuring element member.

In [23] the use of mathematical morphology in the L*a*b* colour model is discussed. A total lexicographical order on the colour vectors is imposed using a weighting function based on an electrostatic potentional. This weighting function assigns a lower weight to colour vectors near the colours with maximum chroma, and higher weights to colour vectors near the lightness axis.

In [4] the distance-based and lexicographical-based approaches are generalised, in order to propose an algorithmic framework allowing the extension of morphological operators to colour images for different colour representations (e.g. RGB, HSL and L*a*b*) and metric distances to a reference colour. The proposed approach is a combination of reduced and conditional ordering: the reduced ordering is based on the distance to a reference colour, e.g. in the L*a*b* colour model the perceptual difference between two colours is given by their Euclidean distance and as reference one can choose white, followed by a lexicographical ordering used to resolve any ambiguities, e.g. in L*a*b*: first the L*-component, followed by the a*-component and then the b*-component. And so standard morphological colour operators are derived.

Having presented these orders, which ordering is best applicable will depend on the practical image analysis tasks. The choice depends mainly on the properties of the images to be processed and on the information the user wants to extract from these images.

4.3.2 New HSV and L*a*b* Colour Vector Ordering

In the RGB colour model

On the RGB cube in figure 4.7 [7] we see that colours lying close to black are ‘dark’ colours while colours lying close to white are ‘light’ colours. We can observe the

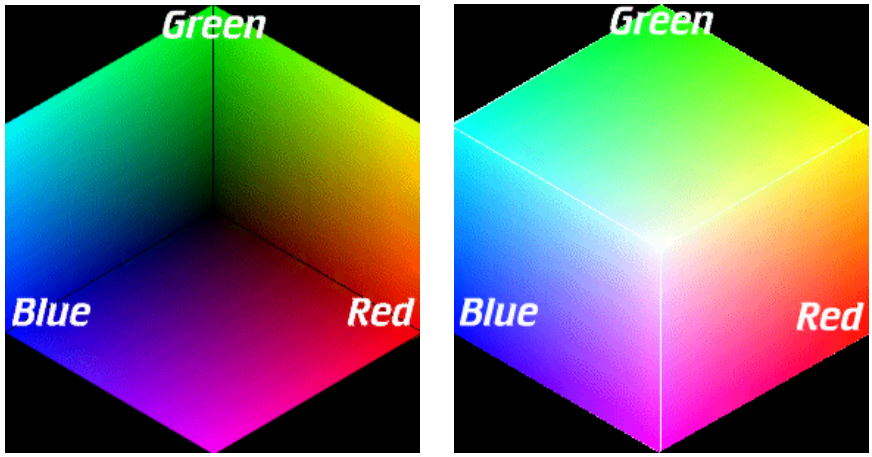


Figure 4.7: Representation of the RGB colour model.

colour hue red for example (we can also choose green or blue). If we start at the white top (with coordinates $(1,1,1)$) and go along the diagonal to the red top $(1,0,0)$ and from there on along the edge to the black top $(0,0,0)$, we see that we go from ‘light’ red over the most ‘bright’ colour red to ‘dark’ red. Inspired by this observation we will sort the colours in the RGB colour model from ‘dark’ colours (close to black) to ‘light’ colours (close to white), with respect to their distance to black and white. So we can define three relations $R_<$, $R_>$ and $R_=$ on RGB, given, for all colours $c(r_c, g_c, b_c)$ and $c'(r_{c'}, g_{c'}, b_{c'})$ in RGB, by

$$
\begin{aligned}
(c,c') \in R_< \quad &\Leftrightarrow \quad d(c,\mathrm{Bl}) < d(c',\mathrm{Bl}) \text{ or} \\
&\quad\; (d(c,\mathrm{Bl}) = d(c',\mathrm{Bl}) \text{ and } d(c,\mathrm{Wh}) > d(c',\mathrm{Wh})) \\
&\Leftrightarrow \quad (c \text{ lies strict closer to black than } c') \text{ or } (c \text{ lies as far} \\
&\quad\; \text{from black as } c' \text{ and } c \text{ lies strict farther from white than } c') \\
(c,c') \in R_> \quad &\Leftrightarrow \quad d(c,\mathrm{Wh}) < d(c',\mathrm{Wh}) \text{ or} \\
&\quad\; (d(c,\mathrm{Wh}) = d(c',\mathrm{Wh}) \text{ and } d(c,\mathrm{Bl}) > d(c',\mathrm{Bl})) \\
&\Leftrightarrow \quad (c \text{ lies strict closer to white than } c') \text{ or } (c \text{ lies as far} \\
&\quad\; \text{from white as } c' \text{ and } c \text{ lies strict farther from black than } c') \\
(c,c') \in R_= \quad &\Leftrightarrow \quad (d(c,\mathrm{Bl}) = d(c',\mathrm{Bl})) \text{ and } (d(c,\mathrm{Wh}) = d(c',\mathrm{Wh})),
\end{aligned}
$$

with d the Euclidean distance, i.e., $d(c,\mathrm{Bl}) = \sqrt{(r_c-0)^2 + (g_c-0)^2 + (b_c-0)^2}$.

1. With the relation $R_<$ colours are first ordered from vectors with smallest distance to black to vectors with largest distance to black. The smaller the distance to black, the lower the colour is ranked. This way the RGB cube is sliced into different parts of spheres around the black top. Colours that are part of the same sphere (around the black top) are then ordered according to their distance with respect to white, from colours with largest distance to white to colours with smallest distance to white. So we will 'cut' the spheres around the black top with spheres with the white top as centre.
2. With the relation $R_>$ we look at the distance with respect to white to know which one of two colours is ranked highest in the RGB colour model. The colour with the smallest distance to white is ordered higher than the other colour. If the distance to white is equal, i.e., if both colours lie on the same sphere centred in the white top, we select that colour lying farthest from black. Again, the RGB cube is sliced into parts of spheres, but now first towards the white top and then towards the black top.
3. Finally in the relation $R_=$ we combine both relations $R_<$ and $R_>$ to say that colours that have the same distance to the black top and the same distance to the white top, and thus lie on a circle (as profile of two spheres) in the RGB cube, are ranked equally.

Figure 4.8 shows how the RGB cube is sliced into spheres around the black top and white top.

Inspired by our idea to look at the black top to determine the 'darkest' colour, and thus the smallest colour, and to look at the white top to determine the 'lightest' colour, and thus the largest colour, we have investigated the cases HSV and L*a*b*.

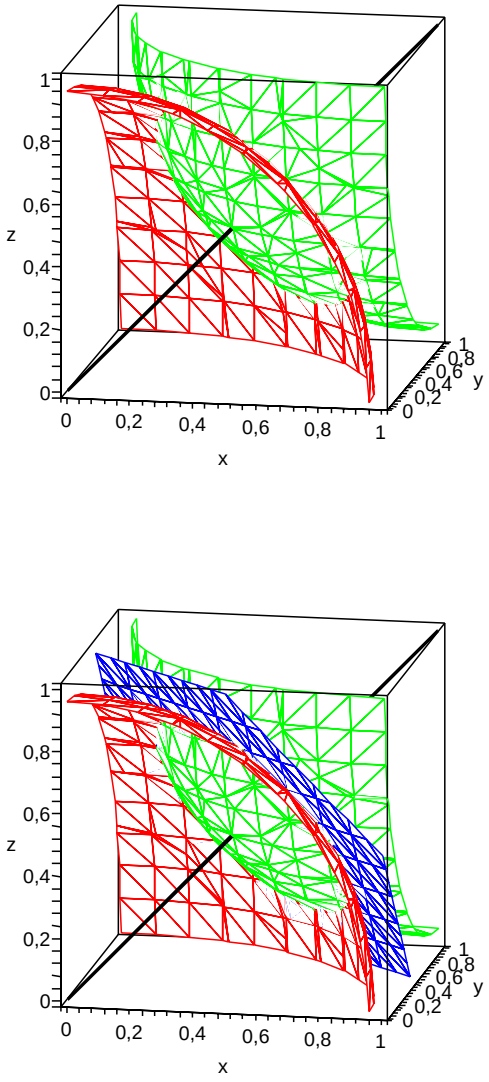


Figure 4.8: At the top: the RGB cube is first sliced into different parts of spheres around the black top for the relation $R_<$ or around the white top for the relation $R_>$ and then respectively cut with spheres with the white top or the black top as centre, at the bottom: for the relation $R_=$ we combine the relations $R_<$ and $R_>$ so that colours lie on a circle.

In the HSV colour model

We denote a colour c in the HSV colour model as $c(h_c, s_c, v_c)$, with $h_c \in [0, 2\pi]$ and $s_c, v_c \in [0, 1]$. Remark that when c is a shade of grey, the hue component h_c is not defined, and we will put $h_c = 0$.

We want to order the colour vectors in the HSV colour model with respect to black and white, so we get:

1. Because the value component V of each colour in HSV gives us the 'grey level' of that colour, we can first order colours by looking at their V-value. A large V-value for a colour means that the colour lies closer to white than to black, and is thus a 'light' colour, while a colour with a small V-value lies closer to black than to white, and is thus a 'dark' colour. The smaller the value component, the smaller the colour is seen.

2. If the value component is equal, we look at the saturation component S of both colours. An S-value of 1 indicates that the colour is completely saturated and contains no white light, i.e., the colour is pure. An S-value of 0 indicates that the colour is a grey value. If $V > 1/2$, we sort the colours from colours with largest S-value to colours with smallest S-value, because the S-value of white is equal to 0 and in this part of the cone the colours are lying close to white and the larger the S-value, the less white light is present in the colour, so the 'darker' the colour will be. If $V < 1/2$, we reverse the order of the S-value and sort the colours from colours with smallest S-value to colours with largest S-value, because the S-value of black is also equal to 0 and in this part of the cone the colours are lying close to black. If for two colours $V = 1/2$, we look at the saturation component and the hue component H to rank these colours. Because we want our ordering to be compatible with the complement co, see section 4.3.4, we have considered the S-value together with the cosinus and sinus of the hue angle.

3. Finally, if two colours have the same V-value, with $V \neq 1/2$, and the same S-value, we look at the hue component H to order these colours. All colour hues are considered to be equally important so that we really have to choose one out of these two colours to be the smallest (or largest) colour. Therefore we have introduced an ordering $\leq_h$ to rank the hue angles.

This gives us an ordering $\leq_{HSV}$ of colour vectors in the HSV colour model, defined for two colours $c(h_c, s_c, v_c)$ and $c'(h_{c'}, s_{c'}, v_{c'})$, as:

$$
\begin{aligned}
c <_{HSV} c' \quad \Leftrightarrow \quad & v_c < v_{c'} \text{ or} \\
& (v_c = v_{c'} > 1/2 \text{ and } s_c > s_{c'}) \text{ or} \\
& (v_c = v_{c'} < 1/2 \text{ and } s_c < s_{c'}) \text{ or} \\
& (v_c = v_{c'} = 1/2 \text{ and } s_c \cos(h_c) < s_{c'} \cos(h_{c'})) \text{ or} \\
& (v_c = v_{c'} = 1/2 \text{ and } s_c \cos(h_c) = s_{c'} \cos(h_{c'}) \text{ and} \\
& \quad s_c \sin(h_c) < s_{c'} \sin(h_{c'})) \text{ or}
\end{aligned}
$$

$$
\begin{array}{rcl}
 & & (v_c = v_{c'} \neq 1/2 \text{ and } s_c = s_{c'} \text{ and } h_c <_h h_{c'}) \\
c >_{HSV} c' & \Leftrightarrow & c' <_{HSV} c \\
 & \Leftrightarrow & v_c > v_{c'} \text{ or} \\
 & & (v_c = v_{c'} > 1/2 \text{ and } s_c < s_{c'}) \text{ or} \\
 & & (v_c = v_{c'} < 1/2 \text{ and } s_c > s_{c'}) \text{ or} \\
 & & (v_c = v_{c'} = 1/2 \text{ and } s_c \cos(h_c) > s_{c'} \cos(h_{c'})) \text{ or} \\
 & & (v_c = v_{c'} = 1/2 \text{ and } s_c \cos(h_c) = s_{c'} \cos(h_{c'}) \text{ and} \\
 & & \quad s_c \sin(h_c) > s_{c'} \sin(h_{c'})) \text{ or} \\
 & & (v_c = v_{c'} \neq 1/2 \text{ and } s_c = s_{c'} \text{ and } h_c >_h h_{c'}) \\
c =_{HSV} c' & \Leftrightarrow & (v_c = v_{c'} \text{ and } s_c = s_{c'} \text{ and } h_c = h_{c'}) \\
c \leq_{HSV} c' & \Leftrightarrow & c <_{HSV} c' \text{ or } c =_{HSV} c',
\end{array}
$$

where $\leq_h = <_h \cup =$, with $<_h$ defined as

if: $((h_c \in [0, \pi[$ and $h_{c'} \in [0, \pi[)$ or $(h_c \in [0, \pi[$ and $h_{c'} \in [\pi, 2\pi[))$ and $h_c < h_{c'}$

then: $h_c <_h h_{c'}$

if: $h_c \in [\pi, 2\pi[$ and $h_{c'} \in [\pi, 2\pi[$ and $2\pi - h_c < 2\pi - h_{c'}$ (or thus $h_c > h_{c'}$)

then: $h_c <_h h_{c'}$.

Properties of $\leq_h$

We know show that

$$(\forall c, c' \in HSV)(h_c \leq_h h_{c'} \text{ and } h_{c'} \leq_h h_c \Rightarrow h_c = h_{c'})$$

Proof

Suppose that $h_c \neq h_{c'}$.

1) From $(h_c \in [0, \pi[$ and $h_{c'} \in [0, \pi[)$ and $h_c <_h h_{c'}$ it follows that $h_c < h_{c'}$.
From $(h_c \in [0, \pi[$ and $h_{c'} \in [0, \pi[)$ and $h_{c'} <_h h_c$ it follows that $h_{c'} < h_c$, and hence a contradiction.

2) From $(h_c \in [0, \pi[$ and $h_{c'} \in [\pi, 2\pi[)$ and $h_c <_h h_{c'}$ it follows that $h_c < h_{c'}$.
The case $(h_c \in [0, \pi[$ and $h_{c'} \in [\pi, 2\pi[)$ and $h_{c'} <_h h_c$ is impossible.

3) From $(h_c \in [\pi, 2\pi[$ and $h_{c'} \in [\pi, 2\pi[)$ and $h_c <_h h_{c'}$ it follows that $h_c > h_{c'}$.
From $(h_c \in [\pi, 2\pi[$ and $h_{c'} \in [\pi, 2\pi[)$ and $h_{c'} <_h h_c$ it follows that $h_{c'} > h_c$, and hence a contradiction.

$\Rightarrow h_c = h_{c'}$.

■

And what is more

$$(\forall c, c', c'' \in HSV)(h_c \leq_h h_{c'} \text{ and } h_{c'} \leq_h h_{c''} \Rightarrow h_c \leq_h h_{c''})$$

Proof

Let $h_c = h_{c'}$ and $h_{c'} \leq_h h_{c''}$, then it holds that $h_c \leq_h h_{c''}$.
Suppose that $h_c \neq h_{c'}$.

1) ($h_c \in [0, \pi[$ and $h_{c'} \in [0, \pi[$) and $h_c < h_{c'}$

   1.1) From $h_{c''} \in [0, \pi[$ and $h_{c'} \leq_h h_{c''}$ it follows that $h_{c'} \leq h_{c''}$.
   $\Rightarrow h_c < h_{c''}$

   1.2) From $h_{c''} \in [\pi, 2\pi[$ and $h_{c'} \leq_h h_{c''}$ it follows that $h_{c'} \leq h_{c''}$
   $\Rightarrow h_c < h_{c''}$

2) ($h_c \in [0, \pi[$ and $h_{c'} \in [\pi, 2\pi[$) and $h_c < h_{c'}$

   2.1) $h_{c''} \in [0, \pi[$, impossible

   2.2) From $h_{c''} \in [\pi, 2\pi[$ and $h_c \in [0, \pi[$ it follows that $h_c < h_{c''}$

3) ($h_c \in [\pi, 2\pi[$ and $h_{c'} \in [\pi, 2\pi[$) and $h_c > h_{c'}$

   3.1) $h_{c''} \in [0, \pi[$, impossible

   3.2) From $h_{c''} \in [\pi, 2\pi[$ and $h_{c'} \leq_h h_{c''}$ it follows that $h_{c'} \geq h_{c''}$
   $\Rightarrow h_c > h_{c''}$

$\Rightarrow h_c \leq_h h_{c''}$.

■

Properties of $\leq_{HSV}$

We examine some properties of our new ordering $\leq_{HSV}$.

1. Reflexive: $(\forall a \in HSV)(a \leq_{HSV} a)$. OK.

2. Antisymmetric: $(\forall a, b \in HSV)(a \leq_{HSV} b \text{ and } b \leq_{HSV} a \stackrel{?}{\Rightarrow} a =_{HSV} b)$

Proof

Suppose that $a \neq_{HSV} b$.

1) $v_a < v_b$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow:

(a) $v_b < v_a$, a contradiction

(b) $v_a = v_b$, a contradiction

2) $v_a = v_b > \frac{1}{2}$ and $s_a > s_b$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow:

(a) $s_b > s_a$, a contradiction

(b) $s_a = s_b$, a contradiction

3) $v_a = v_b < \frac{1}{2}$ and $s_a < s_b$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow:

(a) $s_b < s_a$, a contradiction

(b) $s_a = s_b$, a contradiction

4) $v_a = v_b = \frac{1}{2}$ and $s_a \cos(h_a) < s_b \cos(h_b)$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow:

(a) $s_b \cos(h_b) < s_a \cos(h_a)$, a contradiction

(b) $s_a \cos(h_a) = s_b \cos(h_b)$, a contradiction

5) $v_a = v_b = \frac{1}{2}$ and $s_a \cos(h_a) = s_b \cos(h_b)$ and $s_a \sin(h_a) < s_b \sin(h_b)$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow: $s_b \sin(h_b) < s_a \sin(h_a)$, a contradiction

6) $v_a = v_b \neq \frac{1}{2}$ and $s_a = s_b$ and $h_a <_h h_b$ and $b <_{HSV} a$

From the definition of $<_{HSV}$ would follow: $h_b <_h h_a$, a contradiction

$\Rightarrow$ $v_a = v_b$ and $s_a = s_b$ and $h_a = h_b$

$\Rightarrow$ $a =_{HSV} b$.

■

3. Transitive: $(\forall a, b, c \in HSV)(a \leq_{HSV} b \text{ and } b \leq_{HSV} c \overset{?}{\Rightarrow} a \leq_{HSV} c)$

Proof

Let $a =_{HSV} b$, and $b \leq_{HSV} c$, then it holds that $a \leq_{HSV} c$.
So suppose that $a \neq_{HSV} b$.

1) $v_a < v_b$ and $b \leq_{HSV} c$.

From the definition of $<_{HSV}$ would follow: $v_b \leq v_c$

$\Rightarrow\ v_a < v_c$

$\Rightarrow\ a \leq_{HSV} c$.

2) $v_a = v_b > \frac{1}{2}$ and $s_a > s_b$ and $b \leq_{HSV} c$

From the definition of $<_{HSV}$ would follow:

$(a)\ v_b < v_c$

$\Rightarrow\ v_a < v_c$

$(b)\ v_b = v_c$ and $s_b \geq s_c$

$\Rightarrow\ v_a = v_c$ and $s_a > s_c$

$\Rightarrow\ a \leq_{HSV} c$.

3) $v_a = v_b < \frac{1}{2}$ and $s_a < s_b$ and $b \leq_{HSV} c$

From the definition of $<_{HSV}$ would follow:

$(a)\ v_b < v_c$

$\Rightarrow\ v_a < v_c$

$(b)\ v_b = v_c$ and $s_b \leq s_c$

$\Rightarrow\ v_a = v_c$ and $s_a < s_c$

$\Rightarrow\ a \leq_{HSV} c$.

4) $v_a = v_b = \frac{1}{2}$ and $s_a \cos(h_a) < s_b \cos(h_b)$ and $b \leq_{HSV} c$

From the definition of $<_{HSV}$ would follow:

$(a)\ v_b < v_c$

$\Rightarrow\ v_a < v_c$

$(b)\ v_b = v_c$ and $s_b \cos(h_b) \leq s_c \cos(h_c)$

$\Rightarrow\ v_a = v_c$ and $s_a \cos(h_a) < s_c \cos(h_c)$

$\Rightarrow\ a \leq_{HSV} c$.

5) $v_a = v_b = \frac{1}{2}$ and $s_a \cos(h_a) = s_b \cos(h_b)$ and $s_a \sin(h_a) < s_b \sin(h_b)$ and $b \leq_{HSV} c$

From the definition of $<_{HSV}$ would follow:

(a) $v_b < v_c$

$\Rightarrow$ $v_a < v_c$

(b) $v_b = v_c$ and $s_b \cos(h_b) < s_c \cos(h_c)$

$\Rightarrow$ $v_a = v_c$ and $s_a \cos(h_a) < s_c \cos(h_c)$

(c) $v_b = v_c$ and $s_b \cos(h_b) = s_c \cos(h_c)$ and $s_b \sin(h_b) \leq s_c \sin(h_c)$

$\Rightarrow$ $v_a = v_c$ and $s_a \cos(h_a) = s_c \cos(h_c)$ and $s_a \sin(h_a) < s_c \sin(h_c)$

$\Rightarrow$ $a \leq_{HSV} c$.

6) $v_a = v_b \neq \frac{1}{2}$ and $s_a = s_b$ and $h_a <_h h_b$ and $b \leq_{HSV} c$

From the definition of $<_{HSV}$ would follow:

(a) $v_b < v_c$

$\Rightarrow$ $v_a < v_c$

(b) $v_b = v_c > \frac{1}{2}$ and $s_b > s_c$

$\Rightarrow$ $v_a = v_c$ and $s_a > s_c$

or

(b') $v_b = v_c < \frac{1}{2}$ and $s_b < s_c$

$\Rightarrow$ $v_a = v_c$ and $s_a < s_c$

(c) $v_b = v_c \neq \frac{1}{2}$ and $s_b = s_c$ and $h_b \leq_h h_c$

$\Rightarrow$ $v_a = v_c \neq \frac{1}{2}$ and $s_a = s_c$ and $h_a <_h h_c$

$\Rightarrow$ $a \leq_{HSV} c$.

■

In the L*a*b* and L*u*v* colour model

Here we will only consider the L*a*b* colour model, but the same reasoning can be done for the L*u*v* colour model. A colour c in the L*a*b* colour model can be represented as $c(L^*_c, a^*_c, b^*_c)$, with $L^*_c \in [0, 1]$ and $a^*_c, b^*_c \in [-1, 1]$.

When we order the colours in the L*a*b* colour model by looking at black and white (just as in the RGB colour model), we can first consider the lightness component of the colours, in the same way as described for the value component in the HSV colour model. Secondly, we calculate the hue and chroma of the colours (by converting the rectangular axes a* and b* into polar coordinates)

$$\begin{aligned} h^*_{ab} &= \arctan(b^*/a^*) \quad \text{(hue)} \\ C^*_{ab} &= \sqrt{a^{*2} + b^{*2}} \quad \text{(chroma).} \end{aligned}$$

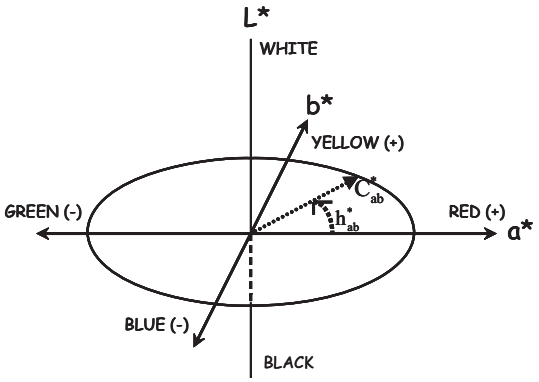


Figure 4.9: Hue and chroma in a graphical representation of the L*a*b* colour model.

In figure 4.9 the hue and chroma are shown in a graphical representation of the L*a*b* colour model. Chroma is defined as the colourfulness of an area judged as a proportion of the brightness of a similarly illuminated reference white [58]. Hue is not defined for shades of grey, but we will put the hue of a shade of grey equal to zero. The scales h^* and C^* together with the lightness L^* correspond to perceptual colour appearance. Analogous with the saturation component in the HSV colour model we can order colours with the same L^*-value according to their C^*-component and h^*-component. If the colours have the same L^*-value, with $L^* \neq 1/2$, and the same C^*-value, then we look at the h^*-value to rank these colours. Again, we have defined a new ordering $\leq_{h^*}$ to order colours w.r.t. their h^*-values.
Summarized, a new colour ordering $\leq_{L^*a^*b^*}$ in L*a*b* is defined as

$$
\begin{aligned}
c <_{L^*a^*b^*} c' \quad \Leftrightarrow \quad & L^*_c < L^*_{c'} \text{ or} \\
& (L^*_c = L^*_{c'} > 1/2 \text{ and } C^*_c > C^*_{c'}) \text{ or} \\
& (L^*_c = L^*_{c'} < 1/2 \text{ and } C^*_c < C^*_{c'}) \text{ or} \\
& (L^*_c = L^*_{c'} = 1/2 \text{ and } C^*_c \cos(h^*_c) < C^*_{c'} \cos(h^*_{c'})) \text{ or} \\
& (L^*_c = L^*_{c'} = 1/2 \text{ and } C^*_c \cos(h^*_c) = C^*_{c'} \cos(h^*_{c'}) \text{ and} \\
& \quad C^*_c \sin(h^*_c) < C^*_{c'} \sin(h^*_{c'})) \text{ or} \\
& (L^*_c = L^*_{c'} \neq 1/2 \text{ and } C^*_c = C^*_{c'} \text{ and } h^*_c <_{h^*} h^*_{c'}) \\
c >_{L^*a^*b^*} c' \quad \Leftrightarrow \quad & c' <_{L^*a^*b^*} c \\
\Leftrightarrow \quad & L^*_c > L^*_{c'} \text{ or} \\
& (L^*_c = L^*_{c'} > 1/2 \text{ and } C^*_c < C^*_{c'}) \text{ or} \\
& (L^*_c = L^*_{c'} < 1/2 \text{ and } C^*_c > C^*_{c'}) \text{ or} \\
& (L^*_c = L^*_{c'} = 1/2 \text{ and } C^*_c \cos(h^*_c) > C^*_{c'} \cos(h^*_{c'})) \text{ or} \\
& (L^*_c = L^*_{c'} = 1/2 \text{ and } C^*_c \cos(h^*_c) = C^*_{c'} \cos(h^*_{c'}) \text{ and} \\
& \quad C^*_c \sin(h^*_c) > C^*_{c'} \sin(h^*_{c'})) \text{ or} \\
& (L^*_c = L^*_{c'} \neq 1/2 \text{ and } C^*_c = C^*_{c'} \text{ and } h^*_c >_{h^*} h^*_{c'})
\end{aligned}
$$

$$c =_{L^*a^*b^*} c' \quad \Leftrightarrow \quad (L^*_c = L^*_{c'} \text{ and } C^*_c = C^*_{c'} \text{ and } h^*_c = h^*_{c'}),$$
$$c \leq_{L^*a^*b^*} c' \quad \Leftrightarrow \quad c <_{L^*a^*b^*} c' \text{ or } c =_{L^*a^*b^*} c',$$

for two colours $c(L^*_c, a^*_c, b^*_c)$ and $c'(L^*_{c'}, a^*_{c'}, b^*_{c'})$ or equivalent $c(L^*_c, C^*_c, h^*_c)$ and $c'(L^*_{c'}, C^*_{c'}, h^*_{c'})$, where $\leq_{h^*} = <_{h^*} \cup =$, with $<_{h^*}$ defined as

if: $((h^*_c \in [0, \pi[$ and $h^*_{c'} \in [0, \pi[)$ or $(h^*_c \in [0, \pi[$ and $h^*_{c'} \in [\pi, 2\pi[))$ and $h^*_c < h^*_{c'}$

then: $h^*_c <_{h^*} h^*_{c'}$

if: $h^*_c \in [\pi, 2\pi[$ and $h^*_{c'} \in [\pi, 2\pi[$ and $2\pi - h^*_c < 2\pi - h^*_{c'}$ (or thus $h^*_c > h^*_{c'}$)

then: $h^*_c <_{h^*} h^*_{c'}$.

Properties of $\leq_{h^*}$

We now show that

$$(\forall c, c' \in L^*a^*b^*)(h^*_c \leq_{h^*} h^*_{c'} \text{ and } h^*_{c'} \leq_{h^*} h^*_c \Rightarrow h^*_c = h^*_{c'})$$

Proof

Suppose that $h^*_c \neq h^*_{c'}$.

1) From $(h^*_c \in [0, \pi[$ and $h^*_{c'} \in [0, \pi[)$ and $h^*_c <_{h^*} h^*_{c'}$ it follows that $h^*_c < h^*_{c'}$.
From $(h^*_c \in [0, \pi[$ and $h^*_{c'} \in [0, \pi[)$ and $h^*_{c'} <_{h^*} h^*_c$ it follows that $h^*_{c'} < h^*_c$, and hence a contradiction.

2) From $(h^*_c \in [0, \pi[$ and $h^*_{c'} \in [\pi, 2\pi[)$ and $h^*_c <_{h^*} h^*_{c'}$ it follows that $h^*_c < h^*_{c'}$.
The case $(h^*_c \in [0, \pi[$ and $h^*_{c'} \in [\pi, 2\pi[)$ and $h^*_{c'} <_{h^*} h^*_c$ is impossible.

3) From $(h^*_c \in [\pi, 2\pi[$ and $h^*_{c'} \in [\pi, 2\pi[)$ and $h^*_c <_{h^*} h^*_{c'}$ it follows that $h^*_c > h^*_{c'}$.
From $(h^*_c \in [\pi, 2\pi[$ and $h^*_{c'} \in [\pi, 2\pi[)$ and $h^*_{c'} <_{h^*} h^*_c$ it follows that $h^*_{c'} > h^*_c$, and hence a contradiction.

$\Rightarrow h^*_c = h^*_{c'}$.

■

And what is more,

$$(\forall c, c', c'' \in L^*a^*b^*)(h^*_c \leq_{h^*} h^*_{c'} \text{ and } h^*_{c'} \leq_{h^*} h^*_{c''} \Rightarrow h^*_c \leq_{h^*} h^*_{c''})$$

Proof

Let $h^*_c = h^*_{c'}$ and $h^*_{c'} \leq_{h^*} h^*_{c''}$, then it holds that $h^*_c \leq_{h^*} h^*_{c''}$.
Suppose that $h^*_c \neq h^*_{c'}$.

1) ($h^*_c \in [0, \pi[$ and $h^*_{c'} \in [0, \pi[$) and $h^*_c < h^*_{c'}$

   1.1) From $h^*_{c''} \in [0, \pi[$ and $h^*_{c'} \leq_{h^*} h^*_{c''}$ it follows that $h^*_{c'} \leq h^*_{c''}$
   $\Rightarrow h^*_c < h^*_{c''}$

   1.2) From $h^*_{c''} \in [\pi, 2\pi[$ and $h^*_{c'} \leq_{h^*} h^*_{c''}$ it follows that $h^*_{c'} \leq h^*_{c''}$
   $\Rightarrow h^*_c < h^*_{c''}$

2) ($h^*_c \in [0, \pi[$ and $h^*_{c'} \in [\pi, 2\pi[$) and $h^*_c < h^*_{c'}$

   2.1) $h^*_{c''} \in [0, \pi[$, impossible

   2.2) From $h^*_{c''} \in [\pi, 2\pi[$ and $h^*_c \in [0, \pi[$ it follows that $h^*_c < h^*_{c''}$

3) ($h^*_c \in [\pi, 2\pi[$ and $h^*_{c'} \in [\pi, 2\pi[$) and $h^*_c > h^*_{c'}$

   3.1) $h^*_{c''} \in [0, \pi[$, impossible

   3.2) From $h^*_{c''} \in [\pi, 2\pi[$ and $h^*_{c'} \leq_{h^*} h^*_{c''}$ it follows that $h^*_{c'} \geq h^*_{c''}$
   $\Rightarrow h^*_c > h^*_{c''}$

$\Rightarrow h^*_c \leq_{h^*} h^*_{c''}$.

■

Properties of $\leq_{L^*a^*b^*}$

We examine some properties of our new ordering $\leq_{L^*a^*b^*}$.

1. Reflexive: $(\forall \alpha \in L^*a^*b^*)(\alpha \leq_{L^*a^*b^*} \alpha)$. OK.

2. Antisymmetric: $(\forall \alpha, \beta \in L^*a^*b^*)(\alpha \leq_{L^*a^*b^*} \beta$ and $\beta \leq_{L^*a^*b^*} \alpha \stackrel{?}{\Rightarrow} \alpha =_{L^*a^*b^*} \beta)$

Proof

Suppose that $\alpha \neq_{L^*a^*b^*} \beta$.

1) $L^*_\alpha < L^*_\beta$ and $\beta <_{L^*a^*b^*} \alpha$

   From the definition of $<_{L^*a^*b^*}$ would follow:

   (a) $L^*_\beta < L^*_\alpha$, a contradiction

   (b) $L^*_\alpha = L^*_\beta$, a contradiction

2) $L^*_\alpha = L^*_\beta > \frac{1}{2}$ and $C^*_\alpha > C^*_\beta$ and $\beta <_{L^*a^*b^*} \alpha$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $C^*_\beta > C^*_\alpha$, a contradiction

(b) $C^*_\alpha = C^*_\beta$, a contradiction

3) $L^*_\alpha = L^*_\beta < \frac{1}{2}$ and $C^*_\alpha < C^*_\beta$ and $\beta <_{L^*a^*b^*} \alpha$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $C^*_\beta < C^*_\alpha$, a contradiction

(b) $C^*_\alpha = C^*_\beta$, a contradiction

4) $L^*_\alpha = L^*_\beta = \frac{1}{2}$ and $C^*_\alpha \cos(h^*_\alpha) < C^*_\beta \cos(h^*_\beta)$ and $\beta <_{L^*a^*b^*} \alpha$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $C^*_\beta \cos(h^*_\beta) < C^*_\alpha \cos(h^*_\alpha)$, a contradiction

(b) $C^*_\alpha \cos(h^*_\alpha) = C^*_\beta \cos(h^*_\beta)$, a contradiction

5) $L^*_\alpha = L^*_\beta = \frac{1}{2}$ and $C^*_\alpha \cos(h^*_\alpha) = C^*_\beta \cos(h^*_\beta)$ and $C^*_\alpha \sin(h^*_\alpha) < C^*_\beta \sin(h^*_\beta)$ and $\beta <_{L^*a^*b^*} \alpha$

From the definition of $<_{HSV}$ would follow: $C^*_\beta \sin(h^*_\beta) < C^*_\alpha \sin(h^*_\alpha)$, a contradiction

6) $L^*_\alpha = L^*_\beta \neq \frac{1}{2}$ and $C^*_\alpha = C^*_\beta$ and $h^*_\alpha <_{h^*} h^*_\beta$ and $\beta <_{L^*a^*b^*} \alpha$

From the definition of $<_{L^*a^*b^*}$ would follow: $h^*_\beta <_{h^*} h^*_\alpha$, a contradiction

$\Rightarrow$ $L^*_\alpha = L^*_\beta$ and $C^*_\alpha = C^*_\beta$ and $h^*_\alpha = h^*_\beta$

$\Rightarrow$ $\alpha =_{L^*a^*b^*} \beta$.

■

3. Transitive: $(\forall \alpha, \beta, \gamma \in L^*a^*b^*)(\alpha \leq_{L^*a^*b^*} \beta$ and $\beta \leq_{L^*a^*b^*} \gamma \stackrel{?}{\Rightarrow} \alpha \leq_{L^*a^*b^*} \gamma)$

Proof

Let $\alpha =_{L^*a^*b^*} \beta$ and $\beta \leq_{L^*a^*b^*} \gamma$, then it holds that $\alpha \leq_{L^*a^*b^*} \gamma$.
Suppose that $\alpha \neq_{L^*a^*b^*} \beta$.

1) $L^*_\alpha < L^*_\beta$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow: $L^*_\beta \leq L^*_\gamma$

$\Rightarrow$ $L^*_\alpha < L^*_\gamma$

$\Rightarrow\ \alpha \leq_{L^*a^*b^*} \gamma$.

2) $L^*_\alpha = L^*_\beta > \frac{1}{2}$ and $C^*_\alpha > C^*_\beta$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $L^*_\beta < L^*_\gamma$

$\Rightarrow\ L^*_\alpha < L^*_\gamma$

(b) $L^*_\beta = L^*_\gamma$ and $C^*_\beta \geq C^*_\gamma$

$\Rightarrow\ L^*_\alpha = L^*_\gamma$ and $C^*_\alpha > C^*_\gamma$

$\Rightarrow\ \alpha \leq_{L^*a^*b^*} \gamma$.

3) $L^*_\alpha = L^*_\beta < \frac{1}{2}$ and $C^*_\alpha < C^*_\beta$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $L^*_\beta < L^*_\gamma$

$\Rightarrow\ L^*_\alpha < L^*_\gamma$

(b) $L^*_\beta = L^*_\gamma$ and $C^*_\beta \leq C^*_\gamma$

$\Rightarrow\ L^*_\alpha = L^*_\gamma$ and $C^*_\alpha < C^*_\gamma$

$\Rightarrow\ \alpha \leq_{L^*a^*b^*} \gamma$.

4) $L^*_\alpha = L^*_\beta = \frac{1}{2}$ and $C^*_\alpha \cos(h^*_\alpha) < C^*_\beta \cos(h^*_\beta)$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $L^*_\beta < L^*_\gamma$

$\Rightarrow\ L^*_\alpha < L^*_\gamma$

(b) $L^*_\beta = L^*_\gamma$ and $C^*_\beta \cos(h^*_\beta) \leq C^*_\gamma \cos(h^*_\gamma)$

$\Rightarrow\ L^*_\alpha < L^*_\gamma$ and $C^*_\alpha \cos(h^*_\alpha) < C^*_\gamma \cos(h^*_\gamma)$

$\Rightarrow\ \alpha \leq_{L^*a^*b^*} \gamma$.

5) $L^*_\alpha = L^*_\beta = \frac{1}{2}$ and $C^*_\alpha \cos(h^*_\alpha) = C^*_\beta \cos(h^*_\beta)$ and $C^*_\alpha \sin(h^*_\alpha) < C^*_\beta \sin(h^*_\beta)$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $L^*_\beta < L^*_\gamma$

$\Rightarrow\ L^*_\alpha < L^*_\gamma$

(b) $L^*_\beta = L^*_\gamma$ and $C^*_\beta \cos(h^*_\beta) < C^*_\gamma \cos(h^*_\gamma)$

$\Rightarrow\ L^*_\alpha = L^*_\gamma$ and $C^*_\alpha \cos(h^*_\alpha) < C^*_\gamma \cos(h^*_\gamma)$

(c) $L^*_\beta = L^*_\gamma$ and $C^*_\beta \cos(h^*_\beta) = C^*_\gamma \cos(h^*_\gamma)$ and $C^*_\beta \sin(h^*_\beta) \leq C^*_\gamma \sin(h^*_\gamma)$

$\Rightarrow$ $L^*_\alpha = L^*_\gamma$ and $C^*_\alpha \cos(h^*_\alpha) = C^*_\gamma \cos(h^*_\gamma)$ and $C^*_\alpha \sin(h^*_\alpha) < C^*_\gamma \sin(h^*_\gamma)$

$\Rightarrow$ $\alpha \leq_{L^*a^*b^*} \gamma$.

6) $L^*_\alpha = L^*_\beta \neq \frac{1}{2}$ and $C^*_\alpha = C^*_\beta$ and $h^*_\alpha <_{h^*} h^*_\beta$ and $\beta \leq_{L^*a^*b^*} \gamma$

From the definition of $<_{L^*a^*b^*}$ would follow:

(a) $L^*_\beta < L^*_\gamma$

$\Rightarrow$ $L^*_\alpha < L^*_\gamma$

(b) $L^*_\beta = L^*_\gamma > \frac{1}{2}$ and $C^*_\beta > C^*_\gamma$

$\Rightarrow$ $L^*_\alpha = L^*_\gamma$ and $C^*_\alpha > C^*_\gamma$

or

(b') $L^*_\beta = L^*_\gamma < \frac{1}{2}$ and $C^*_\beta < C^*_\gamma$

$\Rightarrow$ $L^*_\alpha = L^*_\gamma$ and $C^*_\alpha < C^*_\gamma$

(c) $L^*_\beta = L^*_\gamma \neq \frac{1}{2}$ and $C^*_\beta = C^*_\gamma$ and $h^*_\beta \leq_{h^*} h^*_\gamma$

$\Rightarrow$ $L^*_\alpha = L^*_\gamma \neq \frac{1}{2}$ and $C^*_\alpha = C^*_\gamma$ and $h^*_\alpha <_{h^*} h^*_\gamma$

$\Rightarrow$ $\alpha \leq_{L^*a^*b^*} \gamma$.

■

4.3.3 Associated Minimum and Maximum Operators

Based on the vector ordering for colours introduced in the previous section in the HSV and L*a*b* colour model, we now define new minimum and maximum operators.

The HSV colour model

The poset $(HSV, \leq_{HSV})$ is a totally ordered set; for every two colours c and c' in HSV it holds, by definition of the order relation $\leq_{HSV}$, that $c \leq_{HSV} c'$ or $c' \leq_{HSV} c$, so that $\min_{HSV}(c, c') \in$ HSV and $\max_{HSV}(c, c') \in$ HSV for all $c, c' \in$ HSV.

The minimum (maximum) of a set S of n colours $c_1(h_1, s_1, v_1), \ldots, c_n(h_n, s_n, v_n)$ in HSV is the colour $c_\alpha \in S$ wherefore $c_\alpha \leq_{HSV} c_i$ ($c_\alpha \geq_{HSV} c_i$), for all $i = 1 \ldots n$.

The L*a*b* colour model

The poset $(L^*a^*b^*, \leq_{L^*a^*b^*})$ is a totally ordered set; for every two colours c and c' in L*a*b* it holds, by definition of the order relation $\leq_{L^*a^*b^*}$, that $c \leq_{L^*a^*b^*} c'$ or $c' \leq_{L^*a^*b^*} c$, so that $\min_{L^*a^*b^*}(c, c') \in L^*a^*b^*$ and $\max_{L^*a^*b^*}(c, c') \in L^*a^*b^*$ for all $c, c' \in L^*a^*b^*$.

The minimum (maximum) of a set S of n colours $c_1(L_1^*, a_1^*, b_1^*), \ldots, c_n(L_n^*, a_n^*, b_n^*)$ in L*a*b* is the colour $c_\alpha \in S$ wherefore $c_\alpha \leq_{L^*a^*b^*} c_i$ ($c_\alpha \geq_{L^*a^*b^*} c_i$), for all $i = 1 \ldots n$.

4.3.4 New $(+)$, $(-)$ and $(*)$ Operations between Colours

Apart from a colour ordering, minimum and maximum operators, we also have to define the operations $+$ and $-$ between two colours in order to extend the morphological operators to colour images.

• For the definition of the sum $c + c'$ of two colours c and c' in the RGB, HSV or L*a*b* colour model, we drew our inspiration from the fact that we want the dilation to suppress dark colours and intensify light colours, and the erosion to suppress light colours and intensify dark colours. In the definition of the u-dilation we see an addition of colours, and in the definition of the u-erosion we see a subtraction of colours. If we add white to a colour, we want the colour to become 'lighter', e.g. adding white to red has to give us 'light' red, while if we add black to a colour, we want the colour to become 'darker', e.g. adding black to red has to give us 'dark' red.
• In the RGB colour model we want the definition of the complement co of a colour to fulfil the property that the RGB and CMY colours are complementary colours, thus $co(\text{red}) = \text{cyan}$, $co(\text{green}) = \text{magenta}$, $co(\text{blue}) = \text{yellow}$ and vice versa. For both the HSV and L*a*b* colour model we took into account the fact that (black, white), (red, green) and (blue, yellow) are opponent colour pairs according to the opponent process theory. And we have defined our complement so that the complement of a shade of grey is again a shade of grey.
• At last we define the difference $c - c'$ between the colours c and c' so that the u-dilation and u-erosion fullfill the duality property (see section 4.3.5).

New $(+)$ and $(-)$ Operations between Colours

In RGB

If $c(r_c, g_c, b_c)$ and $c'(r_{c'}, g_{c'}, b_{c'})$ are two colours in RGB, we define the complement $co(c)$ of c, the sum $c + c'$ of c and c', and the difference $c - c'$ between c and c' as:

- $(co(c))(r, g, b) \stackrel{def}{=} \mathbf{1}_{RGB} - c$ with $r \stackrel{def}{=} 1 - r_c, g \stackrel{def}{=} 1 - g_c, b \stackrel{def}{=} 1 - b_c$;

- $(c +_{RGB} c')(r, g, b)$ with $r \stackrel{def}{=} (r_c + r_{c'})/2, g \stackrel{def}{=} (g_c + g_{c'})/2, b \stackrel{def}{=} (b_c + b_{c'})/2$;
- $c -_{RGB} c' \stackrel{def}{=} co(co(c) +_{RGB} c') = co((\mathbf{1}_{RGB} - c) +_{RGB} c') = \mathbf{1}_{RGB} - ((\mathbf{1}_{RGB} - c) +_{RGB} c') = c +_{RGB} (\mathbf{1}_{RGB} - c') = c +_{RGB} co(c')$.

Notice that $co(co(c)) = \mathbf{1}_{RGB} - (\mathbf{1}_{RGB} - c) = c$.

In HSV

We define the complement co of a colour $c(h_c, s_c, v_c)$ and the operations $+$ and $-$ between two colours $c(h_c, s_c, v_c)$ and $c'(h_{c'}, s_{c'}, v_{c'})$ in HSV as:

- $(co(c))(h, s, v) \stackrel{def}{=} \mathbf{1}_{HSV} - c(h_c, s_c, v_c)$ with

  $h \stackrel{def}{=} h_c, s \stackrel{def}{=} s_c, v \stackrel{def}{=} 1 - v_c$, if c is a shade of grey

  $h \stackrel{def}{=} (h_c + \pi) \bmod 2\pi, s \stackrel{def}{=} s_c, v \stackrel{def}{=} 1 - v_c$, otherwise;

- $(c +_{HSV} c')(h, s, v)$ with

  1. $h \stackrel{def}{=} h_c, s \stackrel{def}{=} (s_c + s_{c'})/2, v \stackrel{def}{=} (v_c + v_{c'})/2$, if c' is a shade of grey (analogous if c is a shade of grey)
  2. $h \stackrel{def}{=} (h_c + h_{c'})/2, s \stackrel{def}{=} (s_c + s_{c'})/2, v \stackrel{def}{=} (v_c + v_{c'})/2$, otherwise;

- $(c -_{HSV} c')(h, s, v) \stackrel{def}{=} co(co(c) +_{HSV} c')(h, s, v)$ with

  1. $h \stackrel{def}{=} (((h_c + \pi) \bmod 2\pi) + \pi) \bmod 2\pi = h_c, s \stackrel{def}{=} \frac{s_c + s_{c'}}{2}$, $v \stackrel{def}{=} 1 - \frac{1 - v_c + v_{c'}}{2} = \frac{v_c + 1 - v_{c'}}{2}$, if c' is a shade of grey (analogous if c is a shade of grey)
  2. $h \stackrel{def}{=} ((\frac{(h_c + \pi) \bmod 2\pi) + h_{c'}}{2} + \pi) \bmod 2\pi, s \stackrel{def}{=} \frac{s_c + s_{c'}}{2}, v \stackrel{def}{=} \frac{v_c + 1 - v_{c'}}{2}$, otherwise.

Notice that $co(co(c))(h, s, v) = \mathbf{1}_{HSV} - (\mathbf{1}_{HSV} - c(h_c, s_c, v_c))$ with

$h \stackrel{def}{=} (((h_c + \pi) \bmod 2\pi) + \pi) \bmod 2\pi = h_c, s \stackrel{def}{=} s_c, v \stackrel{def}{=} 1 - (1 - v_c) = v_c$,

thus $co(co(c))(h, s, v) = c(h_c, s_c, v_c)$.

We now prove that our new ordering $\leq_{HSV}$ is compatible with the complement co:

$$\text{for all } c, c' \text{ in } HSV : c \leq_{HSV} c' \stackrel{?}{\Leftrightarrow} co(c) \geq_{HSV} co(c').$$

Proof

1

$$v_c < v_{c'} \quad \Leftrightarrow \quad 1 - v_c > 1 - v_{c'}.$$

2

$$v_c = v_{c'} > \tfrac{1}{2} \ \wedge \ s_c > s_{c'} \quad \Leftrightarrow \quad 1 - v_c = 1 - v_{c'} < \tfrac{1}{2} \ \wedge \ s_c > s_{c'}.$$

3

$$v_c = v_{c'} < \tfrac{1}{2} \ \wedge \ s_c < s_{c'} \quad \Leftrightarrow \quad 1 - v_c = 1 - v_{c'} > \tfrac{1}{2} \ \wedge \ s_c < s_{c'}.$$

4

$$v_c = v_{c'} = \tfrac{1}{2} \wedge s_c \cos(h_c) < s_{c'} \cos(h_{c'}) \stackrel{?}{\Leftrightarrow} 1 - v_c = 1 - v_{c'} = \tfrac{1}{2} \wedge s_c \cos(h_{co(c)}) > s_{c'} \cos(h_{co(c')}),$$

thus

$$s_c \cos(h_c) < s_{c'} \cos(h_{c'}) \stackrel{?}{\Leftrightarrow} s_c \cos(h_{co(c)}) > s_{c'} \cos(h_{co(c')}).$$

It holds that

$$\begin{aligned} s_c \cos(h_c) < s_{c'} \cos(h_{c'}) \quad &\Leftrightarrow \quad -s_c \cos(h_c) > -s_{c'} \cos(h_{c'}) \\ &\Leftrightarrow \quad s_c \cos((h_c + \pi) \bmod 2\pi) > \\ &\qquad s_{c'} \cos((h_{c'} + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad s_c \cos(h_{co(c)}) > s_{c'} \cos(h_{co(c')}), \end{aligned}$$

where we have used the property that

$$\cos(x + \pi) = \cos(x - \pi) = -\cos(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.

5

$$v_c = v_{c'} = \tfrac{1}{2} \ \wedge \ s_c \cos(h_c) = s_{c'} \cos(h_{c'}) \ \wedge \ s_c \sin(h_c) < s_{c'} \sin(h_{c'}) \stackrel{?}{\Leftrightarrow} 1 - v_c = 1 - v_{c'} = \tfrac{1}{2} \wedge s_c \cos(h_{co(c)}) = s_{c'} \cos(h_{co(c')}) \wedge s_c \sin(h_{co(c)}) > s_{c'} \sin(h_{co(c')}),$$

thus

$$s_c \cos(h_c) = s_{c'} \cos(h_{c'}) \ \wedge \ s_c \sin(h_c) < s_{c'} \sin(h_{c'}) \stackrel{?}{\Leftrightarrow}$$

$$s_c \cos(h_{co(c)}) = s_{c'} \cos(h_{co(c')}) \ \wedge \ s_c \sin(h_{co(c)}) > s_{c'} \sin(h_{co(c')}).$$

It holds that

$$\begin{aligned} s_c \cos(h_c) = s_{c'} \cos(h_{c'}) \quad &\Leftrightarrow \quad -s_c \cos(h_c) = -s_{c'} \cos(h_{c'}) \\ &\Leftrightarrow \quad s_c \cos((h_c + \pi) \bmod 2\pi) = \\ &\qquad s_{c'} \cos((h_{c'} + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad s_c \cos(h_{co(c)}) = s_{c'} \cos(h_{co(c')}) \end{aligned}$$

and

$$\begin{aligned} s_c \sin(h_c) < s_{c'} \sin(h_{c'}) \quad &\Leftrightarrow \quad -s_c \sin(h_c) > -s_{c'} \sin(h_{c'}) \\ &\Leftrightarrow \quad s_c \sin((h_c + \pi) \bmod 2\pi) > \\ &\qquad s_{c'} \sin((h_{c'} + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad s_c \sin(h_{co(c)}) > s_{c'} \sin(h_{co(c')}), \end{aligned}$$

where we have used the property that

$$\sin(x + \pi) = \sin(x - \pi) = -\sin(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.

6

$$v_c = v_{c'} \neq 1/2 \ \wedge \ s_c = s_{c'} \ \wedge \ h_c <_h h_{c'} \overset{?}{\Leftrightarrow} 1 - v_c = 1 - v_{c'} \neq 1/2 \wedge s_c = s_{c'} \ \wedge \ h_{co(c)} >_h h_{co(c')},$$

thus

$$h_c <_h h_{c'} \overset{?}{\Leftrightarrow} h_{co(c)} >_h h_{co(c')}.$$

A. $h_c \in [0, \pi[$

i. $h_{c'} \in [0, \pi[$

From $h_c <_h h_{c'}$ and $h_c \in [0, \pi[$ and $h_{c'} \in [0, \pi[$ it follows that $h_c < h_{c'}$. By definition of *co* we get: $h_{co(c)} = (h_c + \pi) \in [\pi, 2\pi[$ and $h_{co(c')} = (h_{c'} + \pi) \in [\pi, 2\pi[$ and hence from $h_c < h_{c'}$ we get $(h_c + \pi) < (h_{c'} + \pi)$, i.e., $h_{co(c)} >_h h_{co(c')}$. The other direction is completely analogous.

ii. $h_{c'} \in [\pi, 2\pi[$

From $h_c <_h h_{c'}$ and $h_c \in [0, \pi[$ and $h_{c'} \in [\pi, 2\pi[$ it follows that $h_c < h_{c'}$. By definition of *co* we get: $h_{co(c)} = (h_c + \pi) \in [\pi, 2\pi[$ and $h_{co(c')} = (h_{c'} + \pi) \bmod 2\pi = (h_{c'} - \pi) \in [0, \pi[$ and hence from $h_{co(c')} \in [0, \pi[$ and $h_{co(c)} \in [\pi, 2\pi[$ we get $h_{co(c')} < h_{co(c)}$, i.e., $h_{co(c')} <_h h_{co(c)}$. The other direction is completely analogous.

B. $h_c \in [\pi, 2\pi[$

i. $h_{c'} \in [0, \pi[$, impossible

ii. $h_{c'} \in [\pi, 2\pi[$
From $h_c <_h h_{c'}$ and $h_c \in [\pi, 2\pi[$ and $h_{c'} \in [\pi, 2\pi[$ it follows that $h_c > h'_{c'}$. By definition of *co* we get: $h_{co(c)} = (h_c - \pi) \in [0, \pi[$ and $h_{co(c')} = (h_{c'} - \pi) \in [0, \pi[$ and hence from $h_c > h_{c'}$ we get $(h_c - \pi) > (h_{c'} - \pi)$, i.e., $h_{co(c)} >_h h_{co(c')}$. The other direction is completely analogous.

■

In L*a*b*

If $c(L^*_c, C^*_c, h^*_c)$ and $c'(L^*_{c'}, C^*_{c'}, h^*_{c'})$ are two colours in the L*a*b* colour model, then we define the complement co of the colour c and the operations $+$ and $-$ between c and c' as

- $(co(c))(L^*, C^*, h^*) \stackrel{def}{=} \mathbf{1}_{L^*a^*b^*} - c(L^*_c, C^*_c, h^*_c)$ with

  $L^* \stackrel{def}{=} 1 - L^*_c, h^* \stackrel{def}{=} h^*_c, C^* \stackrel{def}{=} C^*_c$, if c is a shade of grey

  $L^* \stackrel{def}{=} 1 - L^*_c, h^* \stackrel{def}{=} (h^*_c + \pi) \bmod 2\pi, C^* \stackrel{def}{=} C^*_c$, otherwise;

- $(c +_{L^*a^*b^*} c')(L^*, C^*, h^*)$ with

  1. $L^* \stackrel{def}{=} (L^*_c + L^*_{c'})/2, h^* \stackrel{def}{=} h^*_c, C^* \stackrel{def}{=} (C^*_c + C^*_{c'})/2$, if c' is a shade of grey (analogous if c is a shade of grey)
  2. $L^* \stackrel{def}{=} (L^*_c + L^*_{c'})/2, h^* \stackrel{def}{=} (h^*_c + h^*_{c'})/2, C^* \stackrel{def}{=} (C^*_c + C^*_{c'})/2$, otherwise;

- $(c -_{L^*a^*b^*} c')(L^*, C^*, h^*) \stackrel{def}{=} co(co(c) +_{L^*a^*b^*} c')(L^*, C^*, h^*)$ with

  1. $h^* \stackrel{def}{=} (((h^*_c + \pi) \bmod 2\pi) + \pi) \bmod 2\pi = h^*_c, C^* \stackrel{def}{=} \frac{C^*_c + C^*_{c'}}{2}$,
     $L^* \stackrel{def}{=} 1 - \frac{1 - L^*_c + L^*_{c'}}{2} = \frac{L^*_c + 1 - L^*_{c'}}{2}$, if c' is a shade of grey (analogous if c is a shade of grey)
  2. $h \stackrel{def}{=} ((\frac{(h^*_c + \pi) \bmod 2\pi) + h^*_{c'}}{2} + \pi) \bmod 2\pi, C^* \stackrel{def}{=} \frac{C^*_c + C^*_{c'}}{2}$,
     $L^* \stackrel{def}{=} \frac{L^*_c + 1 - L^*_{c'}}{2}$, otherwise.

Notice that $co(co(c))(L^*, C^*, h^*) = \mathbf{1}_{L^*a^*b^*} - (\mathbf{1}_{L^*a^*b^*} - c(L^*_c, C^*_c, h^*_c))$ with

$h^* \stackrel{def}{=} (((h^*_c + \pi) \bmod 2\pi) + \pi) \bmod 2\pi = h^*_c, C^* \stackrel{def}{=} C^*_c, L^* \stackrel{def}{=} 1 - (1 - L^*_c) = L^*_c$,

thus $co(co(c))(L^*, C^*, h^*) = c(L^*_c, C^*_c, h^*_c)$.

We now prove that our new ordering $\leq_{L^*a^*b^*}$ is compatible with the complement co:

$$\text{for all } c, c' \text{ in } L^*a^*b^* : c \leq_{L^*a^*b^*} c' \stackrel{?}{\Leftrightarrow} co(c) \geq_{L^*a^*b^*} co(c').$$

Proof

1

$$L_c^* < L_{c'}^* \quad \Leftrightarrow \quad 1 - L_c^* > 1 - L_{c'}^*.$$

2

$$L_c^* = L_{c'}^* > \tfrac{1}{2} \ \wedge \ C_c^* > C_{c'}^* \quad \Leftrightarrow \quad 1 - L_c^* = 1 - L_{c'}^* < \tfrac{1}{2} \ \wedge \ C_c^* > C_{c'}^*.$$

3

$$L_c^* = L_{c'}^* < \tfrac{1}{2} \ \wedge \ C_c^* < C_{c'}^* \quad \Leftrightarrow \quad 1 - L_c^* = 1 - L_{c'}^* > \tfrac{1}{2} \ \wedge \ C_c^* < C_{c'}^*.$$

4

$$L_c^* = L_{c'}^* = \tfrac{1}{2} \wedge C_c^* \cos(h_c^*) < C_{c'}^* \cos(h_{c'}^*) \stackrel{?}{\Leftrightarrow} 1 - L_c^* = 1 - L_{c'}^* = \tfrac{1}{2} \wedge C_c^* \cos(h_{co(c)}^*) > C_{c'}^* \cos(h_{co(c')}^*),$$

thus

$$C_c^* \cos(h_c^*) < C_{c'}^* \cos(h_{c'}^*) \stackrel{?}{\Leftrightarrow} C_c^* \cos(h_{co(c)}^*) > C_{c'}^* \cos(h_{co(c')}^*).$$

It holds that

$$\begin{aligned} C_c^* \cos(h_c^*) < C_{c'}^* \cos(h_{c'}^*) \quad &\Leftrightarrow \quad -C_c^* \cos(h_c^*) > -C_{c'}^* \cos(h_{c'}^*) \\ &\Leftrightarrow \quad C_c^* \cos((h_c^* + \pi) \bmod 2\pi) > \\ &\qquad C_{c'}^* \cos((h_{c'}^* + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad C_c^* \cos(h_{co(c)}^*) > C_{c'}^* \cos(h_{co(c')}^*), \end{aligned}$$

where we have used the property that

$$\cos(x + \pi) = \cos(x - \pi) = -\cos(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.

5

$$\begin{aligned} &L_c^* = L_{c'}^* = \tfrac{1}{2} \ \wedge \ C_c^* \cos(h_c^*) = C_{c'}^* \cos(h_{c'}^*) \ \wedge \ C_c^* \sin(h_c^*) < C_{c'}^* \sin(h_{c'}^*) \\ &\stackrel{?}{\Leftrightarrow} 1 - L_c^* = 1 - L_{c'}^* = \tfrac{1}{2} \wedge C_c^* \cos(h_{co(c)}^*) = C_{c'}^* \cos(h_{co(c')}^*) \wedge \\ &C_c^* \sin(h_{co(c)}^*) > C_{c'}^* \sin(h_{co(c')}^*), \end{aligned}$$

thus

$$C_c^* \cos(h_c^*) = C_{c'}^* \cos(h_{c'}^*) \ \wedge \ C_c^* \sin(h_c^*) < C_{c'}^* \sin(h_{c'}^*) \stackrel{?}{\Leftrightarrow}$$

$$C^*_c \cos(h^*_{co(c)}) = C^*_{c'} \cos(h^*_{co(c')}) \wedge C^*_c \sin(h^*_{co(c)}) > C^*_{c'} \sin(h^*_{co(c')}).$$

It holds that

$$\begin{aligned} C^*_c \cos(h^*_c) = C^*_{c'} \cos(h^*_{c'}) \quad &\Leftrightarrow \quad -C^*_c \cos(h^*_c) = -C^*_{c'} \cos(h^*_{c'}) \\ &\Leftrightarrow \quad C^*_c \cos((h^*_c + \pi) \bmod 2\pi) = \\ &\qquad C^*_{c'} \cos((h^*_{c'} + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad C^*_c \cos(h^*_{co(c)}) = C^*_{c'} \cos(h^*_{co(c')}) \end{aligned}$$

and

$$\begin{aligned} C^*_c \sin(h^*_c) < C^*_{c'} \sin(h^*_{c'}) \quad &\Leftrightarrow \quad -C^*_c \sin(h^*_c) > -C^*_{c'} \sin(h^*_{c'}) \\ &\Leftrightarrow \quad C^*_c \sin((h^*_c + \pi) \bmod 2\pi) > \\ &\qquad C^*_{c'} \sin((h^*_{c'} + \pi) \bmod 2\pi) \\ &\Leftrightarrow \quad C^*_c \sin(h^*_{co(c)}) > C^*_{c'} \sin(h^*_{co(c')}), \end{aligned}$$

where we have used the property that

$$\sin(x + \pi) = \sin(x - \pi) = -\sin(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.

6

$$\begin{aligned} L^*_c = L^*_{c'} \neq 1/2 \ \wedge \ C^*_c = C^*_{c'} \ \wedge \ h^*_c <_{h^*} h^*_{c'} \overset{?}{\Leftrightarrow} 1 - L^*_c = 1 - L^*_{c'} \neq 1/2 \wedge \\ C^*_c = C^*_{c'} \ \wedge \ h^*_{co(c)} >_{h^*} h^*_{co(c')}, \end{aligned}$$

thus

$$h^*_c <_{h^*} h^*_{c'} \overset{?}{\Leftrightarrow} h^*_{co(c)} >_{h^*} h^*_{co(c')}.$$

A. $h^*_c \in [0, \pi[$

i. $h^*_{c'} \in [0, \pi[$

From $h^*_c <_{h^*} h^*_{c'}$ and $h^*_c \in [0, \pi[$ and $h^*_{c'} \in [0, \pi[$ it follows that $h^*_c < h^*_{c'}$. By definition of *co* we get: $h^*_{co(c)} = (h^*_c + \pi) \in [\pi, 2\pi[$ and $h^*_{co(c')} = (h^*_{c'} + \pi) \in [\pi, 2\pi[$ and hence from $h^*_c < h^*_{c'}$ we get $(h^*_c + \pi) < (h^*_{c'} + \pi)$, i.e., $h^*_{co(c)} >_{h^*} h^*_{co(c')}$. The other direction is completely analogous.

ii. $h^*_{c'} \in [\pi, 2\pi[$

From $h^*_c <_{h^*} h^*_{c'}$ and $h^*_c \in [0, \pi[$ and $h^*_{c'} \in [\pi, 2\pi[$ it follows that $h^*_c < h^*_{c'}$. By definition of *co* we get: $h^*_{co(c)} = (h^*_c + \pi) \in [\pi, 2\pi[$ and $h^*_{co(c')} = (h^*_{c'} + \pi) \bmod 2\pi = (h^*_{c'} - \pi) \in [0, \pi[$ and hence from $h^*_{co(c')} \in [0, \pi[$ and $h^*_{co(c)} \in [\pi, 2\pi[$ we get $h^*_{co(c')} < h^*_{co(c)}$, i.e., $h^*_{co(c')} <_{h^*} h^*_{co(c)}$. The other direction is completely analogous.

$B.$ $h_c^* \in [\pi, 2\pi[$

$i.$ $h_{c'}^* \in [0, \pi[$, impossible

$ii.$ $h_{c'}^* \in [\pi, 2\pi[$

From $h_c^* <_{h^*} h_{c'}^*$ and $h_c^* \in [\pi, 2\pi[$ and $h_{c'}^* \in [\pi, 2\pi[$ it follows that $h_c^* > h_{c'}^*$. By definition of co we get: $h_{co(c)}^* = (h_c^* - \pi) \in [0, \pi[$ and $h_{co(c')}^* = (h_{c'}^* - \pi) \in [0, \pi[$ and hence from $h_c^* > h_{c'}^*$ we get $(h_c^* - \pi) > (h_{c'}^* - \pi)$, i.e., $h_{co(c)}^* >_{h^*} h_{co(c')}^*$. The other direction is completely analogous.

■

New ($*$) Operation between Colours

To apply the fuzzy mathematical morphological operators to a colour image in HSV or L*a*b* we also need to define the product $*$ of two colours.

In RGB

Because we consider all three colour components R, G and B to be equally important we will usually take a symmetric greyscale structuring element as 3-dimensional structuring element in RGB, i.e.,

$$B(i,j,1) = B(i,j,2) = B(i,j,3) = \begin{pmatrix} c_{B_1} & c_{B_{11}} & c_{B_1} \\ c_{B_{11}} & 1 & c_{B_{11}} \\ c_{B_1} & c_{B_{11}} & c_{B_1} \end{pmatrix}, \ 1 \leq i,j, \leq 3.$$

According to the distance to the centre pixel we give a certain weight c_{B_1} or $c_{B_{11}}$, thus a certain grade of importance, to each observed colour in the window, where $c_{B_{11}} \geq c_{B_1}$. But we can also choose for example a structuring element of the form

$$B(i,j,1) = \begin{pmatrix} \circ & \bullet & \circ \\ \bullet & 1 & \bullet \\ \circ & \bullet & \circ \end{pmatrix}, B(i,j,2) = B(i,j,3) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, 1 \leq i,j, \leq 3,$$

to give a weight to the R-component only, so that the G- and B-component remain unchanged. We then define the product $*$ of a colour $c(r_c, g_c, b_c)$ and a colour $c_B(r_{c_B}, g_{c_B}, b_{c_B})$ of the chosen structuring element B componentwisely as

$$(c * c_B)(r,g,b) \text{ with } r \stackrel{def}{=} r_c \cdot r_{c_B}, g \stackrel{def}{=} g_c \cdot g_{c_B}, b \stackrel{def}{=} b_c \cdot b_{c_B}.$$

In HSV

To get a 3-dimensional structuring element in the HSV colour model we can transform the structuring element B chosen in RGB into HSV. If we want to define a multiplication $*$ between a colour c in HSV and a colour $c_{B_{HSV}}$ of a structuring element B_{HSV} in HSV, we always have to scale the values of the H-, S- and V-component of the

colour $c_{B_{HSV}}$ to the interval $[0,1]$ (note $c^*_{B_{HSV}}$) to give a weight to the colour c. This way we can also choose immediately a structuring element B^*_{HSV}, for example,

$$B^*_{HSV}(i,j,1) = B^*_{HSV}(i,j,2) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, B^*_{HSV}(i,j,3) = \begin{pmatrix} \circ & \bullet & \circ \\ \bullet & 1 & \bullet \\ \circ & \bullet & \circ \end{pmatrix},$$

$1 \leq i,j, \leq 3$, to attach importance to the value component only. So we define the product $*$ of a colour $c(h_c, s_c, v_c)$ and a colour $c^*_{B_{HSV}}(h_{c^*_{B_{HSV}}}, s_{c^*_{B_{HSV}}}, v_{c^*_{B_{HSV}}})$ of the chosen structuring element B^*_{HSV} as

$$(c * c^*_{B_{HSV}})(h,s,v) \text{ with } h \stackrel{def}{=} h_c \cdot h_{c^*_{B_{HSV}}}, s \stackrel{def}{=} s_c \cdot s_{c^*_{B_{HSV}}}, v \stackrel{def}{=} v_c \cdot v_{c^*_{B_{HSV}}}.$$

Some t-norms on the lattice $(HSV, \leq_{HSV})$ are

$$\mathcal{T}_{\min}(\gamma,\delta) = \min_{HSV}(\gamma,\delta)$$
$$\mathcal{T}_*(\gamma,\delta) \quad = \gamma * \delta, \qquad \forall \gamma, \delta \in HSV.$$

The $\mathcal{S}$-implicators induced by $\mathcal{T}_{\min}$ (and $\mathcal{T}_*$) and the standard negator $\mathcal{N}_s(c) = \mathbf{1}_{HSV} - c$, for all $c \in HSV$, on $(HSV, \leq_{HSV})$ are then given by

$$\mathcal{I}_{\mathcal{T}_{\min},\mathcal{N}_s}(\gamma,\delta) = \max_{HSV}(\mathbf{1}_{HSV} - \gamma, \delta)$$
$$\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s}(\gamma,\delta) \quad = \mathbf{1}_{HSV} - (\gamma * (\mathbf{1}_{HSV} - \delta)), \qquad \forall \gamma, \delta \in HSV.$$

In L*a*b*

In the L*a*b* colour model we proceed analogously as in the HSV colour model. Two t-norms on the lattice $(L^*a^*b^*, \leq_{L^*a^*b^*})$ are $\mathcal{T}_{\min}(\gamma,\delta) = \min_{L^*a^*b^*}(\gamma,\delta)$ and $\mathcal{T}_*(\gamma,\delta) = \gamma * \delta$ defined for all γ and δ in $L^*a^*b^*$. The $\mathcal{S}$-implicators induced by $\mathcal{T}_{\min}$ (and $\mathcal{T}_*$) and the standard negator $\mathcal{N}_s(c) = \mathbf{1}_{L^*a^*b^*} - c$, for all $c \in L^*a^*b^*$, on $(L^*a^*b^*, \leq_{L^*a^*b^*})$ are

$$\mathcal{I}_{\mathcal{T}_{\min},\mathcal{N}_s}(\gamma,\delta) = \max_{L^*a^*b^*}(\mathbf{1}_{L^*a^*b^*} - \gamma, \delta)$$
$$\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s}(\gamma,\delta) \quad = \mathbf{1}_{L^*a^*b^*} - (\gamma * (\mathbf{1}_{L^*a^*b^*} - \delta)), \qquad \forall \gamma, \delta \in L^*a^*b^*.$$

Summarized:

For HSV:

$(HSV, \leq_{HSV})$ is a poset, and what is more, by definition it holds that

$$(\forall c, c' \in HSV)(c \leq_{HSV} c' \text{ or } c' \leq_{HSV} c),$$

so that $(HSV, \leq_{HSV})$ is a totally ordered set ($\max_{HSV}(a,b)$ and $\min_{HSV}(a,b)$ exist for all $a, b \in HSV$), and thus a lattice. The greatest element in $(HSV, \leq_{HSV})$ is $\mathbf{1} = (0,0,1)$ and the smallest element is $\mathbf{0} = (0,0,0)$, so we get a bounded complete lattice. We will sometimes drop the index HSV.

Since colour images in the HSV colour model can be modelled as $\mathbb{R}^2 - (HSV, \leq_{HSV})$ mappings and because $(HSV, \leq_{HSV})$ is a complete lattice, we can identify colour images in HSV with $\mathcal{L}$-fuzzy sets on $\mathbb{R}^2$, with $(\mathcal{L}, \leq_{\mathcal{L}}) = (HSV, \leq_{HSV})$, and thus define for a family $(A_i)_{i=1}^n$ of colour images in HSV

$$\begin{aligned} \bigcap_{i=1}^{n} A_i(x) &= \min_{i=1\ldots n} A_i(x),\ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n} A_i(x) &= \max_{i=1\ldots n} A_i(x),\ \forall x \in \mathbb{R}^2, \end{aligned}$$

so that $(\mathcal{F}_{HSV}(X), \cap_{HSV}, \cup_{HSV})$ is a lattice with the ordering defined for all $A, B \in \mathcal{F}_{HSV}(X)$ as

$$A \subseteq_{HSV} B \Leftrightarrow (\forall x \in \mathbb{R}^2)(A(x) \leq_{HSV} B(x)).$$

Let $\mathcal{N}$ be a negator on HSV, $\mathcal{T}$ a t-norm and $\mathcal{S}$ a t-conorm on HSV. Consider a colour image A in HSV and a family $(A_i)_{i=1}^n$ of colour images in HSV, then we define

$$\begin{aligned} co_{\mathcal{N}} A(x) &= \mathcal{N}(A(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcap_{i=1}^{n}{}_{\mathcal{T}}\ A_i(x) &= \mathcal{T}(A_1(x), A_2(x), \ldots, A_n(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n}{}_{\mathcal{S}}\ A_i(x) &= \mathcal{S}(A_1(x), A_2(x), \ldots, A_n(x)),\ \forall x \in \mathbb{R}^2. \end{aligned}$$

For L*a*b*:

$(L^*a^*b^*, \leq_{L^*a^*b^*})$ is a poset, and what is more, by definition it holds that

$$(\forall c, c' \in L^*a^*b^*)(c \leq_{L^*a^*b^*} c' \text{ or } c' \leq_{L^*a^*b^*} c),$$

so that $(L^*a^*b^*, \leq_{L^*a^*b^*})$ is a totally ordered set ($\max_{L^*a^*b^*}(a,b)$ and $\min_{L^*a^*b^*}(a,b)$ exist for all $a, b \in L^*a^*b^*$), and thus a lattice. The greatest element in $(L^*a^*b^*, \leq_{L^*a^*b^*})$ is $\mathbf{1} = (1,0,0)$ and the smallest element is $\mathbf{0} = (0,0,0)$, so we get a bounded complete lattice. We will sometimes drop the index $L^*a^*b^*$.

Since colour images in the L*a*b* colour model can be modelled as $\mathbb{R}^2 - (L^*a^*b^*, \leq_{L^*a^*b^*})$ mappings and because $(L^*a^*b^*, \leq_{L^*a^*b^*})$ is a complete lattice, we can identify colour images in L*a*b* with $\mathcal{L}$-fuzzy sets on $\mathbb{R}^2$, with

$(\mathcal{L}, \leq_{\mathcal{L}}) = (L^*a^*b^*, \leq_{L^*a^*b^*})$, and thus define for a family $(A_i)_{i=1}^n$ of colour images in $L^*a^*b^*$

$$\begin{aligned} \bigcap_{i=1}^{n} A_i(x) &= \min_{i=1\ldots n} A_i(x),\ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n} A_i(x) &= \max_{i=1\ldots n} A_i(x),\ \forall x \in \mathbb{R}^2, \end{aligned}$$

so that $(\mathcal{F}_{L^*a^*b^*}(X), \cap_{L^*a^*b^*}, \cup_{L^*a^*b^*})$ is a lattice with the ordering defined for all $A, B \in \mathcal{F}_{L^*a^*b^*}(X)$ as

$$A \subseteq_{L^*a^*b^*} B \Leftrightarrow (\forall x \in \mathbb{R}^2)(A(x) \leq_{L^*a^*b^*} B(x)).$$

Let $\mathcal{N}$ be a negator on L*a*b*, $\mathcal{T}$ a t-norm and $\mathcal{S}$ a t-conorm on L*a*b*. For a colour image A in L*a*b* and a family $(A_i)_{i=1}^n$ of colour images in L*a*b* we define

$$\begin{aligned} co_{\mathcal{N}} A(x) &= \mathcal{N}(A(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcap_{i=1}^{n}{}_{\mathcal{T}} A_i(x) &= \mathcal{T}(A_1(x), A_2(x), \ldots, A_n(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n}{}_{\mathcal{S}} A_i(x) &= \mathcal{S}(A_1(x), A_2(x), \ldots, A_n(x)),\ \forall x \in \mathbb{R}^2. \end{aligned}$$

4.3.5 New Vector-based Approach to Colour Morphology

We extend the basic morphological operators dilation and erosion for greyscale images based on the threshold and fuzzy approach to colour images modelled in HSV and L*a*b*. And we have even tried to extend the 'theoretical' u-operators dilation and erosion to useful unambiguous operators for colour images.
We have taken into account the important remark we found in paper [56] that states that a generalisation of morphological operations to complete lattices is necessary for a mathematically coherent application of morphological operators to greyscale images. Any computer implementation of mathematical morphology only works with digital images defined on a finite grid and whose grey values range in a finite interval in $\mathbb{R}$, thus the set of grey values is bounded, so that a computation of grey values can give a value outside this set, in other words, one can get an arithmetic overflow. A careless approach to this problem can lead to operators looking at first sight like dilations and erosions, but which do not have their usual algebraic properties and behave in fact differently. This problem of greylevel overflow can be solved by using complete lattices. And because we want to extend the greyscale morphological operators to morphological operators acting on colour images, so that greyscale morphology becomes a restriction of colour morphology, we have looked for complete lattices in our extension to colour.

Extension of greyscale morphology to colour morphology in HSV

• Threshold approach

Let A be a colour image, represented as a $\mathbb{R}^2 - (HSV, \leq_{HSV})$ mapping, and B a binary structuring element ($\subseteq \mathbb{R}^2$).

Definition 4.13. *Let A be a colour image in HSV and B a binary structuring element. The **threshold 'colour' dilation** $\vec{D}_t(A,B)$ and the **threshold 'colour' erosion** $\vec{E}_t(A,B)$ are the colour images given by*

$$\vec{D}_t(A,B)(y) \stackrel{def}{=}_{HSV} \max_{x \in T_y(B)} A(x) \quad \textit{for } y \in \mathbb{R}^2,$$

$$\vec{E}_t(A,B)(y) \stackrel{def}{=}_{HSV} \min_{x \in T_y(B)} A(x) \quad \textit{for } y \in \mathbb{R}^2.$$

Property 4.14.

$$\vec{D}_t(\overline{\mathbf{0}},B) =_{HSV} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(\overline{\mathbf{1}},B) =_{HSV} \overline{\mathbf{1}}$$
$$\vec{D}_t(A,\emptyset) =_{HSV} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(A,\emptyset) =_{HSV} \overline{\mathbf{1}}.$$

Proof

$$\vec{D}_t(\overline{\mathbf{0}},B)(y) \stackrel{def}{=}_{HSV} \max_{x \in T_y(B)} \overline{\mathbf{0}}(x) =_{HSV} \mathbf{0} =_{HSV} \overline{\mathbf{0}}(y), \ \ \forall y \in \mathbb{R}^2.$$

$$\vec{E}_t(\overline{\mathbf{1}},B)(y) \stackrel{def}{=}_{HSV} \min_{x \in T_y(B)} \overline{\mathbf{1}}(x) =_{HSV} \mathbf{1} =_{HSV} \overline{\mathbf{1}}(y), \ \ \forall y \in \mathbb{R}^2.$$

$$\vec{D}_t(A,\emptyset)(y) \stackrel{def}{=}_{HSV} \max_{x \in T_y(\emptyset)} A(x) =_{HSV} \max \emptyset =_{HSV} \mathbf{0} =_{HSV} \overline{\mathbf{0}}(y), \ \ \forall y \in \mathbb{R}^2.$$

$$\vec{E}_t(A,\emptyset)(y) \stackrel{def}{=}_{HSV} \min_{x \in T_y(\emptyset)} A(x) =_{HSV} \min \emptyset =_{HSV} \mathbf{1} =_{HSV} \overline{\mathbf{1}}(y), \ \ \forall y \in \mathbb{R}^2.$$

■

Property 4.15 (Duality dilation-erosion).

$$\vec{D}_t(A,B) =_{HSV} co(\vec{E}_t(co(A),B))$$
$$\vec{E}_t(A,B) =_{HSV} co(\vec{D}_t(co(A),B)).$$

Proof

$$\begin{aligned}
co(\vec{E}_t(co(A),B))(y) &\stackrel{def}{=}_{HSV} \mathbf{1}_{HSV} - \vec{E}_t(co(A),B)(y)\\
&\stackrel{def}{=}_{HSV} \mathbf{1}_{HSV} - \min_{x\in T_y(B)} co(A)(x)\\
&\stackrel{(*)}{=}_{HSV} \max_{x\in T_y(B)} \mathbf{1}_{HSV} - (\mathbf{1}_{HSV} - A(x))\\
&=_{HSV} \max_{x\in T_y(B)} A(x)\\
&=_{HSV} \vec{D}_t(A,B)(y), \qquad \textit{for all } y\in\mathbb{R}^2.
\end{aligned}$$

$\hookrightarrow (*)$ *For two colours c and c' in HSV we get*

$$\begin{aligned}
\mathbf{1}_{HSV} - \min_{HSV}(c,c') &=_{HSV} \mathbf{1}_{HSV} - c' \qquad (\textit{suppose } c' \leq_{HSV} c)\\
&=_{HSV} \max_{HSV}(\mathbf{1}_{HSV} - c, \mathbf{1}_{HSV} - c').
\end{aligned}$$

$$\begin{aligned}
co(\vec{D}_t(co(A),B))(y) &\stackrel{def}{=}_{HSV} \mathbf{1}_{HSV} - \vec{D}_t(co(A),B)(y)\\
&\stackrel{def}{=}_{HSV} \mathbf{1}_{HSV} - \max_{x\in T_y(B)} co(A)(x)\\
&\stackrel{(*)}{=}_{HSV} \min_{x\in T_y(B)} \mathbf{1}_{HSV} - (\mathbf{1}_{HSV} - A(x))\\
&=_{HSV} \min_{x\in T_y(B)} A(x)\\
&=_{HSV} \vec{E}_t(A,B)(y), \qquad \textit{for all } y\in\mathbb{R}^2.
\end{aligned}$$

$\hookrightarrow (*)$ *For two colours c and c' in HSV we get*

$$\begin{aligned}
\mathbf{1}_{HSV} - \max_{HSV}(c,c') &=_{HSV} \mathbf{1}_{HSV} - c \qquad (\textit{suppose } c' \leq_{HSV} c)\\
&=_{HSV} \min_{HSV}(\mathbf{1}_{HSV} - c, \mathbf{1}_{HSV} - c').
\end{aligned}$$

■

Property 4.16 (Monotonicity). *If A and B are two colour images in HSV, and C and C' are two binary structuring elements, then it holds that*

$$\begin{aligned}
A \subseteq_{HSV} B \quad\Rightarrow\quad & \vec{D}_t(A,C) \subseteq_{HSV} \vec{D}_t(B,C) \textit{ and}\\
& \vec{E}_t(A,C) \subseteq_{HSV} \vec{E}_t(B,C)\\
C \subseteq C' \quad\Rightarrow\quad & \vec{D}_t(A,C) \subseteq_{HSV} \vec{D}_t(A,C') \textit{ and}\\
& \vec{E}_t(A,C) \supseteq_{HSV} \vec{E}_t(A,C').
\end{aligned}$$

Proof

$$\begin{aligned}
A \subseteq_{HSV} B &\Leftrightarrow A(z) \leq_{HSV} B(z), \quad \forall z \in \mathbb{R}^2 \\
&\Rightarrow \max_{z \in T_y(C)} A(z) \leq_{HSV} \max_{z \in T_y(C)} B(z), \quad \forall z \in \mathbb{R}^2 \\
&\Rightarrow \vec{D}_t(A, C) \subseteq_{HSV} \vec{D}_t(B, C), \\
A \subseteq_{HSV} B &\Leftrightarrow A(z) \leq_{HSV} B(z), \quad \forall z \in \mathbb{R}^2 \\
&\Rightarrow \min_{z \in T_y(C)} A(z) \leq_{HSV} \min_{z \in T_y(C)} B(z), \quad \forall z \in \mathbb{R}^2 \\
&\Rightarrow \vec{E}_t(A, C) \subseteq_{HSV} \vec{E}_t(B, C).
\end{aligned}$$

$$\begin{aligned}
C \subseteq C' &\Leftrightarrow (\forall x \in \mathbb{R}^2)(x \in C \Rightarrow x \in C') \\
&\Rightarrow (\forall x \in \mathbb{R}^2)(x \in T_y(C) \Rightarrow x \in T_y(C')), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow T_y(C) \subseteq T_y(C'), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \max_{x \in T_y(C)} A(x) \leq_{HSV} \max_{x \in T_y(C')} A(x), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \vec{D}_t(A, C) \subseteq_{HSV} \vec{D}_t(A, C'), \\
C \subseteq C' &\Leftrightarrow (\forall x \in \mathbb{R}^2)(x \in C \Rightarrow x \in C') \\
&\Rightarrow (\forall x \in \mathbb{R}^2)(x \in T_y(C) \Rightarrow x \in T_y(C')), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow T_y(C) \subseteq T_y(C'), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \min_{x \in T_y(C)} A(x) \geq_{HSV} \min_{x \in T_y(C')} A(x), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \vec{E}_t(A, C) \supseteq_{HSV} \vec{E}_t(A, C').
\end{aligned}$$

■

Property 4.17 (Inclusion).

$$\vec{E}_t(A, B) \subseteq_{HSV} \vec{D}_t(A, B).$$

Proof
By definition of $\leq_{HSV}$ it holds that for all y in $\mathbb{R}^2$

$$\min_{x \in T_y(B)} A(x) \leq_{HSV} \max_{x \in T_y(B)} A(x).$$

■

Property 4.18.

$$\mathbf{0} \in B \quad \Rightarrow \quad A \subseteq_{HSV} \vec{D}_t(A, B) \text{ and } \vec{E}_t(A, B) \subseteq_{HSV} A.$$

Proof

$$
\begin{aligned}
\mathbf{0} \in B \quad &\Rightarrow \quad y \in T_y(B), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad \vec{D}_t(A,B)(y) \stackrel{def}{=}_{HSV} \max_{x \in T_y(B)} A(x) \geq_{HSV} A(y), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad A \subseteq_{HSV} \vec{D}_t(A,B), \\
\mathbf{0} \in B \quad &\Rightarrow \quad y \in T_y(B), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad \vec{E}_t(A,B)(y) \stackrel{def}{=}_{HSV} \min_{x \in T_y(B)} A(x) \leq_{HSV} A(y), \quad \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad \vec{E}_t(A,B) \subseteq_{HSV} A.
\end{aligned}
$$

■

Property 4.19 (Interaction with intersection and union). *Consider a family $(A_i)_{i=1}^n$ of colour images in HSV and a family $(B_i)_{i=1}^n$ of binary structuring elements. For the t-'colour' dilation it holds that*

$$
\begin{aligned}
\vec{D}_t(\bigcap_{i=1}^n A_i, B) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_t(A_i,B) \\
\vec{D}_t(A, \bigcap_{i=1}^n B_i) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_t(A,B_i); \\
\vec{D}_t(\bigcup_{i=1}^n A_i, B) &=_{HSV} \bigcup_{i=1}^n \vec{D}_t(A_i,B) \\
\vec{D}_t(A, \bigcup_{i=1}^n B_i) &=_{HSV} \bigcup_{i=1}^n \vec{D}_t(A,B_i).
\end{aligned}
$$

Proof

1. $(\forall j \in \{1,\ldots,n\})(\bigcap_{i=1}^n A_i \subseteq_{HSV} A_j)$

   $\stackrel{4.16}{\Rightarrow} (\forall j \in \{1,\ldots,n\})(\vec{D}_t(\bigcap_{i=1}^n A_i, B) \subseteq_{HSV} \vec{D}_t(A_j,B))$

   $\Rightarrow \vec{D}_t(\bigcap_{i=1}^n A_i, B) \subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_t(A_i,B).$

2. $(\forall j \in \{1,\ldots,n\})(\bigcap_{i=1}^n B_i \subseteq_{HSV} B_j)$

   $\stackrel{4.16}{\Rightarrow} (\forall j \in \{1,\ldots,n\})(\vec{D}_t(A, \bigcap_{i=1}^n B_i) \subseteq_{HSV} \vec{D}_t(A,B_j))$

   $\Rightarrow \vec{D}_t(A, \bigcap_{i=1}^n B_i) \subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_t(A,B_i).$

3.

$$\begin{aligned}
\vec{D}_t(\bigcup_{i=1}^{n} A_i, B)(y) &=_{HSV} \max_{x\in T_y(B)} (\bigcup_{i=1}^{n} A_i)(x) \\
&=_{HSV} \max_{x\in T_y(B)} \max_{i=1\ldots n} A_i(x) \\
&=_{HSV} \max_{i=1\ldots n} \max_{x\in T_y(B)} A_i(x) \\
&=_{HSV} \bigcup_{i=1}^{n} \vec{D}_t(A_i, B)(y), \qquad \forall y \in \mathbb{R}^2.
\end{aligned}$$

4.

$$\begin{aligned}
\vec{D}_t(A, \bigcup_{i=1}^{n} B_i)(y) &=_{HSV} \max_{x\in T_y(\cup_{i=1}^{n} B_i)} A(x) \\
&\overset{(*)}{=}_{HSV} \max_{x\in \cup_{i=1}^{n} T_y(B_i)} A(x) \\
&=_{HSV} \max_{i=1\ldots n} \max_{x\in T_y(B_i)} A(x) \\
&=_{HSV} \bigcup_{i=1}^{n} \vec{D}_t(A, B_i)(y), \quad \forall y \in \mathbb{R}^2.
\end{aligned}$$

$\hookrightarrow (*)$

$$\begin{aligned}
T_y(\cup_{i=1}^{n} B_i) &= \{x \in \mathbb{R}^2 \mid x - y \in \cup_{i=1}^{n} B_i\} \\
&= \{x \in \mathbb{R}^2 \mid x - y \in B_1 \vee \ldots \vee x - y \in B_n\} \\
&= \{x \in \mathbb{R}^2 | x - y \in B_1\} \cup \ldots \cup \{x \in \mathbb{R}^2 | x - y \in B_n\} \\
&= \cup_{i=1}^{n} T_y(B_i)
\end{aligned}$$

■

For the t-'colour' erosion it holds that

$$\begin{aligned}
\vec{E}_t(\bigcap_{i=1}^{n} A_i, B) &=_{HSV} \bigcap_{i=1}^{n} \vec{E}_t(A_i, B) \\
\vec{E}_t(A, \bigcap_{i=1}^{n} B_i) &\supseteq_{HSV} \bigcup_{i=1}^{n} \vec{E}_t(A, B_i); \\
\vec{E}_t(\bigcup_{i=1}^{n} A_i, B) &\supseteq_{HSV} \bigcup_{i=1}^{n} \vec{E}_t(A_i, B) \\
\vec{E}_t(A, \bigcup_{i=1}^{n} B_i) &=_{HSV} \bigcap_{i=1}^{n} \vec{E}_t(A, B_i).
\end{aligned}$$

Proof

1.

$$\begin{aligned}
\vec{E}_t(\bigcap_{i=1}^{n} A_i, B)(y) &=_{HSV} \min_{x\in T_y(B)} (\bigcap_{i=1}^{n} A_i)(x) \\
&=_{HSV} \min_{x\in T_y(B)} \min_{i=1\ldots n} A_i(x) \\
&=_{HSV} \min_{i=1\ldots n} \min_{x\in T_y(B)} A_i(x) \\
&=_{HSV} \bigcap_{i=1}^{n} \vec{E}_t(A_i, B)(y), \qquad \forall y \in \mathbb{R}^2.
\end{aligned}$$

2. $(\forall j \in \{1,\ldots,n\})(\bigcap_{i=1}^{n} B_i \subseteq_{HSV} B_j)$

$\overset{4.16}{\Rightarrow} (\forall j \in \{1,\ldots,n\})(\vec{E}_t(A, \bigcap_{i=1}^{n} B_i) \supseteq_{HSV} \vec{E}_t(A, B_j))$

$\Rightarrow \vec{E}_t(A, \bigcap_{i=1}^{n} B_i) \supseteq_{HSV} \bigcup_{i=1}^{n} \vec{E}_t(A, B_i).$

3. $(\forall j \in \{1,\ldots,n\})(A_j \subseteq_{HSV} \bigcup_{i=1}^{n} A_i)$

$\overset{4.16}{\Rightarrow} (\forall j \in \{1,\ldots,n\})(\vec{E}_t(A_j, B) \subseteq_{HSV} \vec{E}_t(\bigcup_{i=1}^{n} A_i, B))$

$\Rightarrow \bigcup_{i=1}^{n} \vec{E}_t(A_i, B) \subseteq_{HSV} \vec{E}_t(\bigcup_{i=1}^{n} A_i, B).$

4.

$$\begin{aligned}
\vec{E}_t(A, \bigcup_{i=1}^{n} B_i)(y) &=_{HSV} \min_{x\in T_y(\cup_{i=1}^{n} B_i)} A(x) \\
&=_{HSV} \min_{x\in \cup_{i=1}^{n} T_y(B_i)} A(x) \\
&=_{HSV} \min_{i=1\ldots n} \min_{x\in T_y(B_i)} A(x) \\
&=_{HSV} \bigcap_{i=1}^{n} \vec{E}_t(A, B_i)(y), \quad \forall y \in \mathbb{R}^2.
\end{aligned}$$

■

Property 4.20. *Let A be a colour image in HSV and B and C two binary structuring elements, then*

$$\begin{aligned}
\vec{D}_t(\vec{D}_t(A, B), C) &=_{HSV} \vec{D}_t(\vec{D}_t(A, C), B) \\
\vec{E}_t(\vec{E}_t(A, B), C) &=_{HSV} \vec{E}_t(\vec{E}_t(A, C), B).
\end{aligned}$$

Proof
On the one hand we have

$$\begin{aligned}\vec{D}_t(\vec{D}_t(A,B),C)(y) &=_{HSV} \max_{x\in T_y(C)} \vec{D}_t(A,B)(x)\\ &=_{HSV} \max_{x\in T_y(C)} \max_{z\in T_x(B)} A(z)\\ &=_{HSV} \max\{A(z) \mid (\exists x \in T_y(C))(z \in T_x(B))\}, \forall y \in \mathbb{R}^2.\end{aligned}$$

And on the other hand we get

$$\begin{aligned}\vec{D}_t(\vec{D}_t(A,C),B)(y) &=_{HSV} \max_{x\in T_y(B)} \vec{D}_t(A,C)(x)\\ &=_{HSV} \max_{x\in T_y(B)} \max_{z\in T_x(C)} A(z)\\ &=_{HSV} \max\{A(z) \mid (\exists x \in T_y(B))(z \in T_x(C))\}, \forall y \in \mathbb{R}^2.\end{aligned}$$

So

$$\begin{aligned}(\exists x \in T_y(C))(z \in T_x(B)) &\overset{?}{\Leftrightarrow} (\exists x \in T_y(B))(z \in T_x(C))\\ (\exists x \in \mathbb{R}^2)(x-y \in C \wedge z-x \in B) &\overset{?}{\Leftrightarrow} (\exists x \in \mathbb{R}^2)(x-y \in B \wedge z-x \in C).\end{aligned}$$

Put $p = z+y-x$, then $x-y = z-p$ and $z-x = p-y$, thus

$$(\exists x \in \mathbb{R}^2)(x-y \in C \wedge z-x \in B) \overset{!}{\Leftrightarrow} (\exists p \in \mathbb{R}^2)(z-p \in B \wedge p-y \in C).$$

The other expression can be proven analogously.

■

• Fuzzy approach

The support d_A of a colour image A in HSV is defined as the set

$$d_A = \{x \in \mathbb{R}^2 \mid A(x) >_{HSV} \mathbf{0}\}.$$

Definition 4.21. *Let A be a colour image and B a colour structuring element (both seen as $(HSV, \leq_{HSV})$-fuzzy sets), $\mathcal{C}$ a conjunctor and $\mathcal{I}$ an implicator on $(HSV, \leq_{HSV})$. The **fuzzy 'colour' dilation** $\vec{D}_{\mathcal{C}}(A,B)$ and the **fuzzy 'colour' erosion** $\vec{E}_{\mathcal{I}}(A,B)$ are the $(HSV, \leq_{HSV})$-fuzzy sets defined as*

$$\begin{aligned}\vec{D}_{\mathcal{C}}(A,B)(y) &\overset{def}{=}_{HSV} \max_{x\in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) \quad \text{for } y \in \mathbb{R}^2,\\ \vec{E}_{\mathcal{I}}(A,B)(y) &\overset{def}{=}_{HSV} \min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) \quad \text{for } y \in \mathbb{R}^2.\end{aligned}$$

Notice that we can write

$$\begin{aligned}\max_{x\in T_y(d_B)} \mathcal{C}(B(x-y),A(x)) &=_{HSV} \max_{x\in\mathbb{R}^2} \mathcal{C}(R_B(x,y),A(x)),\\ \min_{x\in T_y(d_B)} \mathcal{I}(B(x-y),A(x)) &=_{HSV} \min_{x\in\mathbb{R}^2} \mathcal{I}(R_B(x,y),A(x)),\end{aligned}$$

for all y in $\mathbb{R}^2$, whereby

$$R_B : \mathbb{R}^2 \times \mathbb{R}^2 \to (HSV, \leq_{HSV}),$$

$R_B = B \circ V$ with V defined as $V(x,y) = x - y = (x_1, x_2) - (y_1, y_2) = (x_1 - y_1, x_2 - y_2),\ \forall (x,y) \in (\mathbb{R}^2)^2$ so that $R_B(x,y) = B(x-y)$.

With every structuring element B we can associate a $(HSV, \leq_{HSV})$-fuzzy relation R_B on $\mathbb{R}^2$. We will now prove some properties of the new fuzzy colour morphological operators. Some of them can be deduced from the properties of $\mathcal{L}$-fuzzy relational images, proved in [12]. The only difference is that here we do not always require that the conjunctor $\mathcal{C}$ is a triangular norm.

Property 4.22. *[12] Let $\mathcal{T}$ be a t-norm and $\mathcal{I}$ be an implicator on $(HSV, \leq_{HSV})$, then it holds:*

$$\vec{D}_{\mathcal{T}}(\overline{\mathbf{0}}, B) =_{HSV} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{1}}, B) =_{HSV} \overline{\mathbf{1}}.$$

If $B(\mathbf{0}) =_{HSV} \mathbf{1}$, then it holds:

$$\vec{D}_{\mathcal{T}}(\overline{\mathbf{1}}, B) =_{HSV} \overline{\mathbf{1}} \quad \textit{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{0}}, B) =_{HSV} \overline{\mathbf{0}}.$$

Proof
For all y in $\mathbb{R}^2$:

$$\max_{x\in T_y(d_B)} \mathcal{T}(B(x-y), \overline{\mathbf{0}}(x)) \overset{def}{=} \max_{x\in T_y(d_B)} \mathcal{T}(B(x-y), \mathbf{0}) \overset{2.15}{=} \mathbf{0} = \overline{\mathbf{0}}(y),$$

$$\min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), \overline{\mathbf{1}}(x)) \overset{def}{=} \min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), \mathbf{1}) \overset{2.16}{=} \mathbf{1} = \overline{\mathbf{1}}(y).$$

If $B(\mathbf{0}) =_{HSV} \mathbf{1}$, then we get for all $y \in \mathbb{R}^2$

$$\max_{x\in T_y(d_B)} \mathcal{T}(B(x-y), \mathbf{1}) \geq_{HSV} \mathcal{T}(B(y-y), \mathbf{1}) = \mathcal{T}(\mathbf{1}, \mathbf{1}) = 1 = \overline{\mathbf{1}}(y),$$

$$\min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), \mathbf{0}) \leq_{HSV} \mathcal{I}(B(y-y), \mathbf{0}) = \mathcal{I}(\mathbf{1}, \mathbf{0}) = \mathbf{0} = \overline{\mathbf{0}}(y).$$

■

Property 4.23 (Duality dilation-erosion). *[12] Let $\mathcal{T}$ be a t-norm on HSV, $\mathcal{N}$ an involutive negator on HSV and $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ the corresponding $\mathcal{S}$-implicator. For every colour image A and colour structuring element B we have*

$$\begin{aligned} co_{\mathcal{N}}\vec{D_{\mathcal{T}}}(A,B) &=_{HSV} \vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(co_{\mathcal{N}}A,B) \\ \vec{D_{\mathcal{T}}}(co_{\mathcal{N}}A,B) &=_{HSV} co_{\mathcal{N}}(\vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(A,B)). \end{aligned}$$

Property 4.24 (Monotonicity). *[Generalisation of [12]] If A and B are two colour images, C and C' two colour structuring elements, $\mathcal{C}_1$ and $\mathcal{C}_2$ two conjunctors and $\mathcal{I}_1$ and $\mathcal{I}_2$ two implicators on HSV, then it holds that*

$$\begin{aligned} A \subseteq_{HSV} B \quad &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{HSV} \vec{D_{\mathcal{C}}}(B,C) \text{ and} \\ &\qquad\quad \vec{E_{\mathcal{I}}}(A,C) \subseteq_{HSV} \vec{E_{\mathcal{I}}}(B,C) \\ C \subseteq_{HSV} C' \quad &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{HSV} \vec{D_{\mathcal{C}}}(A,C') \text{ and} \\ &\qquad\quad \vec{E_{\mathcal{I}}}(A,C) \supseteq_{HSV} \vec{E_{\mathcal{I}}}(A,C') \\ \mathcal{C}_1 \subseteq_{HSV} \mathcal{C}_2 \quad &\Rightarrow \quad \vec{D_{\mathcal{C}_1}}(A,C) \subseteq_{HSV} \vec{D_{\mathcal{C}_2}}(A,C) \\ \mathcal{I}_1 \subseteq_{HSV} \mathcal{I}_2 \quad &\Rightarrow \quad \vec{E_{\mathcal{I}_1}}(A,C) \subseteq_{HSV} \vec{E_{\mathcal{I}_2}}(A,C) \end{aligned}$$

Proof

$$\begin{aligned} A \subseteq_{HSV} B \quad &\Leftrightarrow \quad A(x) \leq_{HSV} B(x), \quad \forall x \in \mathbb{R}^2 \\ &\Rightarrow \quad \mathcal{C}(C(x-y),A(x)) \leq_{HSV} \mathcal{C}(C(x-y),B(x)), \quad \forall x,y \in \mathbb{R}^2 \\ &\qquad\qquad \textit{($\mathcal{C}$ is increasing)} \\ &\Rightarrow \quad \max_{x \in T_y(d_C)} \mathcal{C}(C(x-y),A(x)) \leq_{HSV} \\ &\qquad\qquad \max_{x \in T_y(d_C)} \mathcal{C}(C(x-y),B(x)), \ \forall y \in \mathbb{R}^2 \\ &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{HSV} \vec{D_{\mathcal{C}}}(B,C). \end{aligned}$$

Analogously for $A \subseteq_{HSV} B \Rightarrow \vec{E_{\mathcal{I}}}(A,C) \subseteq_{HSV} \vec{E_{\mathcal{I}}}(B,C)$.

$$\begin{aligned} C \subseteq_{HSV} C' \quad &\Leftrightarrow \quad C(x-y) \leq_{HSV} C'(x-y), \quad \forall x,y \in \mathbb{R}^2 \\ &\Rightarrow \quad \mathcal{C}(C(x-y),A(x)) \leq_{HSV} \mathcal{C}(C'(x-y),A(x)), \quad \forall x,y \in \mathbb{R}^2 \\ &\qquad\qquad \textit{($\mathcal{C}$ is increasing)} \\ &\Rightarrow \quad \max_{x \in T_y(d_C)} \mathcal{C}(C(x-y),A(x)) \leq_{HSV} \\ &\qquad\qquad \max_{x \in T_y(d_{C'})} \mathcal{C}(C'(x-y),A(x)), \ \forall y \in \mathbb{R}^2 \\ &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{HSV} \vec{D_{\mathcal{C}}}(A,C'). \end{aligned}$$

Analogously for $C \subseteq_{HSV} C' \Rightarrow \vec{E_{\mathcal{I}}}(A,C) \supseteq_{HSV} \vec{E_{\mathcal{I}}}(A,C')$.

$$\begin{aligned}
\mathcal{C}_1 \subseteq_{HSV} \mathcal{C}_2 \quad &\Rightarrow \quad \mathcal{C}_1(B(x-y),A(x)) \leq_{HSV} \mathcal{C}_2(B(x-y),A(x)), \quad \forall x,y \in \mathbb{R}^2 \\
&\Rightarrow \quad \max_{x \in T_y(d_B)} \mathcal{C}_1(B(x-y),A(x)) \leq_{HSV} \\
&\qquad\qquad \max_{x \in T_y(d_B)} \mathcal{C}_2(B(x-y),A(x)), \ \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad \vec{D_{\mathcal{C}_1}}(A,B) \subseteq_{HSV} \vec{D_{\mathcal{C}_2}}(A,B).
\end{aligned}$$

$$\begin{aligned}
\mathcal{I}_1 \subseteq_{HSV} \mathcal{I}_2 \quad &\Rightarrow \quad \mathcal{I}_1(B(x-y),A(x)) \leq_{HSV} \mathcal{I}_2(B(x-y),A(x)), \quad \forall x,y \in \mathbb{R}^2 \\
&\Rightarrow \quad \min_{x \in T_y(d_B)} \mathcal{I}_1(B(x-y),A(x)) \leq_{HSV} \\
&\qquad\qquad \min_{x \in T_y(d_B)} \mathcal{I}_2(B(x-y),A(x)), \ \forall y \in \mathbb{R}^2 \\
&\Rightarrow \quad \vec{E_{\mathcal{I}_1}}(A,B) \subseteq_{HSV} \vec{E_{\mathcal{I}_2}}(A,B).
\end{aligned}$$

■

Property 4.25 (Inclusion). *[Generalisation of [12]] Let* $\mathcal{C}$ *be a seminorm and* $\mathcal{I}$ *an edge-implicator on HSV. Consider a colour image* A *and a 'normalized' colour structuring element* B*, that is,* $(\forall y \in \mathbb{R}^2)(\exists z \in \mathbb{R}^2)(B(z-y) =_{HSV} \mathbf{1})$*. It holds that*

$$\vec{E}_{\mathcal{I}}(A,B) \subseteq_{HSV} \vec{D}_{\mathcal{C}}(A,B).$$

Proof

$$\begin{aligned}
\vec{E}_{\mathcal{I}}(A,B)(y) \quad &\stackrel{def}{=}_{HSV} \quad \min_{x \in T_y(d_B)} \mathcal{I}(B(x-y),A(x)) \\
&\leq_{HSV} \quad \mathcal{I}(B(z-y),A(z)) \\
&=_{HSV} \quad \mathcal{I}(\mathbf{1},A(z)) \\
&=_{HSV} \quad A(z) \\
&=_{HSV} \quad \mathcal{C}(\mathbf{1},A(z)) \\
&=_{HSV} \quad \mathcal{C}(B(z-y),A(z)) \\
&\leq_{HSV} \quad \max_{x \in T_y(d_B)} \mathcal{C}(B(x-y),A(x)) \\
&\stackrel{def}{=}_{HSV} \quad \vec{D}_{\mathcal{C}}(A,B)(y), \quad \forall y \in \mathbb{R}^2.
\end{aligned}$$

■

Property 4.26. *[Generalisation of [12]] Let* $\mathcal{C}$ *be a seminorm and* $\mathcal{I}$ *an edge- implicator on HSV. For every colour image* A *and every colour structuring element* B*, it holds that*

$$B(\mathbf{0}) =_{HSV} \mathbf{1} \quad \Rightarrow \quad A \subseteq_{HSV} \vec{D}_{\mathcal{C}}(A,B) \text{ and } \vec{E}_{\mathcal{I}}(A,B) \subseteq_{HSV} A.$$

Proof

$$\begin{aligned}
\vec{D}_{\mathcal{C}}(A,B)(y) &\stackrel{def}{=}_{HSV} \max_{x\in T_y(d_B)} \mathcal{C}(B(x-y),A(x)) \\
&\geq_{HSV} \mathcal{C}(B(y-y),A(y)) \\
&=_{HSV} \mathcal{C}(\mathbf{1},A(y)) \\
&=_{HSV} A(y), \quad \forall y\in\mathbb{R}^2.
\end{aligned}$$

$$\begin{aligned}
\vec{E}_{\mathcal{I}}(A,B)(y) &\stackrel{def}{=}_{HSV} \min_{x\in T_y(d_B)} \mathcal{I}(B(x-y),A(x)) \\
&\leq_{HSV} \mathcal{I}(B(y-y),A(y)) \\
&=_{HSV} \mathcal{I}(\mathbf{1},A(y)) \\
&=_{HSV} A(y), \quad \forall y\in\mathbb{R}^2.
\end{aligned}$$

■

From property 4.24 it follows that

Property 4.27 (Interaction with intersection and union). *[Generalisation of [12]] Consider a family $(A_i)_{i=1}^n$ of colour images and a family $(B_i)_{i=1}^n$ of colour structuring elements. For the $\mathcal{C}$-'colour' dilation it holds that*

$$\begin{aligned}
\vec{D}_{\mathcal{C}}(\bigcap_{i=1}^n A_i,B) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_{\mathcal{C}}(A_i,B) \\
\vec{D}_{\mathcal{C}}(A,\bigcap_{i=1}^n B_i) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{D}_{\mathcal{C}}(A,B_i); \\
\vec{D}_{\mathcal{C}}(\bigcup_{i=1}^n A_i,B) &\supseteq_{HSV} \bigcup_{i=1}^n \vec{D}_{\mathcal{C}}(A_i,B) \\
\vec{D}_{\mathcal{C}}(A,\bigcup_{i=1}^n B_i) &\supseteq_{HSV} \bigcup_{i=1}^n \vec{D}_{\mathcal{C}}(A,B_i).
\end{aligned}$$

For the $\mathcal{I}$-'colour' erosion it holds that

$$\begin{aligned}
\vec{E}_{\mathcal{I}}(\bigcap_{i=1}^n A_i,B) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{E}_{\mathcal{I}}(A_i,B) \\
\vec{E}_{\mathcal{I}}(A,\bigcap_{i=1}^n B_i) &\supseteq_{HSV} \bigcup_{i=1}^n \vec{E}_{\mathcal{I}}(A,B_i); \\
\vec{E}_{\mathcal{I}}(\bigcup_{i=1}^n A_i,B) &\supseteq_{HSV} \bigcup_{i=1}^n \vec{E}_{\mathcal{I}}(A_i,B) \\
\vec{E}_{\mathcal{I}}(A,\bigcup_{i=1}^n B_i) &\subseteq_{HSV} \bigcap_{i=1}^n \vec{E}_{\mathcal{I}}(A,B_i).
\end{aligned}$$

But we can prove more, i.e.,

$$\begin{aligned}
\vec{D_{\mathcal{C}}}(A, \bigcup_{i=1}^{n} B_i) &=_{HSV} \bigcup_{i=1}^{n} \vec{D_{\mathcal{C}}}(A, B_i) \\
\vec{E_{\mathcal{I}}}(A, \bigcup_{i=1}^{n} B_i) &=_{HSV} \bigcap_{i=1}^{n} \vec{E_{\mathcal{I}}}(A, B_i); \\
\vec{D_{\mathcal{C}}}(\bigcup_{i=1}^{n} A_i, B) &=_{HSV} \bigcup_{i=1}^{n} \vec{D_{\mathcal{C}}}(A_i, B) \\
\vec{E_{\mathcal{I}}}(\bigcap_{i=1}^{n} A_i, B) &=_{HSV} \bigcap_{i=1}^{n} \vec{E_{\mathcal{I}}}(A_i, B).
\end{aligned}$$

Proof

$$\begin{aligned}
\vec{D_{\mathcal{C}}}(A, \bigcup_{i=1}^{n} B_i)(y) &\stackrel{def}{=}_{HSV} \max_{x \in T_y(d_{\cup_{i=1}^{n} B_i})} \mathcal{C}(\bigcup_{i=1}^{n} B_i(x-y), A(x)) \\
&\stackrel{(*)}{=}_{HSV} \max_{x \in T_y(\cup_{i=1}^{n} d_{B_i})} \mathcal{C}(\max_{i=1\ldots n} B_i(x-y), A(x)).
\end{aligned}$$

And since a conjunctor $\mathcal{C}$ is increasing, we get:

$$\begin{aligned}
\vec{D_{\mathcal{C}}}(A, \bigcup_{i=1}^{n} B_i)(y) &=_{HSV} \max_{x \in \cup_{i=1\ldots n} T_y(d_{B_i})} \max_{i=1\ldots n} \mathcal{C}(B_i(x-y), A(x)) \\
&=_{HSV} \max_{i=1\ldots n} (\max_{x \in T_y(d_{B_1})} \mathcal{C}(B_1(x-y), A(x)), \ldots, \\
&\qquad \max_{x \in T_y(d_{B_n})} \mathcal{C}(B_n(x-y), A(x))) \\
&=_{HSV} \bigcup_{i=1}^{n} \vec{D_{\mathcal{C}}}(A, B_i)(y), \quad \forall y \in \mathbb{R}^2.
\end{aligned}$$

$\hookrightarrow (*)$

$$\begin{aligned}
T_y(d_{\cup_{i=1}^{n} B_i}) &= \{x \in \mathbb{R}^2 \mid x - y \in d_{\cup_{i=1}^{n} B_i}\} \\
&= \{x \in \mathbb{R}^2 \mid \cup_{i=1}^{n} B_i(x-y) > 0\} \\
&= \{x \in \mathbb{R}^2 \mid \max_{i=1\ldots n} B_i(x-y) > 0\} \\
&= \{x \in \mathbb{R}^2 \mid B_1(x-y) > 0 \vee \ldots \vee B_n(x-y) > 0\} \\
&= \{x \in \mathbb{R}^2 \mid x - y \in d_{B_1} \vee \ldots \vee x - y \in d_{B_n}\} \\
&= \{x \in \mathbb{R}^2 \mid x - y \in \cup_{i=1}^{n} d_{B_i}\} \\
&= T_y(\cup_{i=1}^{n} d_{B_i})
\end{aligned}$$

Analogously,

$$\begin{aligned}\vec{E_{\mathcal{I}}}(A,\bigcup_{i=1}^{n}B_i)(y) &\stackrel{def}{=}_{HSV} \min_{x\in T_y(d_{\cup_{i=1}^{n}B_i})}\mathcal{I}(\bigcup_{i=1}^{n}B_i(x-y),A(x))\\ &=_{HSV} \min_{x\in T_y(\cup_{i=1}^{n}d_{B_i})}\mathcal{I}(\max_{i=1\ldots n}B_i(x-y),A(x)).\end{aligned}$$

And since an implicator $\mathcal{I}$ is decreasing in its first argument, we get:

$$\begin{aligned}\vec{E_{\mathcal{I}}}(A,\bigcup_{i=1}^{n}B_i)(y) &=_{HSV} \min_{x\in\cup_{i=1}^{n}T_y(d_{B_i})}\min_{i=1\ldots n}\mathcal{I}(B_i(x-y),A(x))\\ &=_{HSV} \min_{i=1\ldots n}(\min_{x\in T_y(d_{B_1})}\mathcal{I}(B_1(x-y),A(x)),\ldots,\\ &\qquad \min_{x\in T_y(d_{B_n})}\mathcal{I}(B_n(x-y),A(x)))\\ &=_{HSV} \bigcap_{i=1}^{n}\vec{E_{\mathcal{I}}}(A,B_i)(y),\quad \forall y\in\mathbb{R}^2.\end{aligned}$$

Analogously,

$$\begin{aligned}\vec{D_{\mathcal{C}}}(\bigcup_{i=1}^{n}A_i,B)(y) &\stackrel{def}{=}_{HSV} \max_{x\in T_y(d_B)}\mathcal{C}(B(x-y),\bigcup_{i=1}^{n}A_i(x))\\ &\stackrel{def}{=}_{HSV} \max_{x\in T_y(d_B)}\mathcal{C}(B(x-y),\max_{i=1\ldots n}A_i(x)).\end{aligned}$$

And since a conjunctor $\mathcal{C}$ is increasing, we get:

$$\begin{aligned}\vec{D_{\mathcal{C}}}(\bigcup_{i=1}^{n}A_i,B)(y) &=_{HSV} \max_{x\in T_y(d_B)}\max_{i=1\ldots n}\mathcal{C}(B(x-y),A_i(x))\\ &=_{HSV} \max_{i=1\ldots n}(\max_{x\in T_y(d_B)}\mathcal{C}(B(x-y),A_1(x)),\ldots,\\ &\qquad \max_{x\in T_y(d_B)}\mathcal{C}(B(x-y),A_n(x)))\\ &=_{HSV} \bigcup_{i=1}^{n}\vec{D_{\mathcal{C}}}(A_i,B)(y),\quad \forall y\in\mathbb{R}^2.\end{aligned}$$

Analogously,

$$\begin{aligned}\vec{E_{\mathcal{I}}}(\bigcap_{i=1}^{n}A_i,B)(y) &\stackrel{def}{=}_{HSV} \min_{x\in T_y(d_B)}\mathcal{I}(B(x-y),\bigcap_{i=1}^{n}A_i(x))\\ &\stackrel{def}{=}_{HSV} \min_{x\in T_y(d_B)}\mathcal{I}(B(x-y),\min_{i=1\ldots n}A_i(x)).\end{aligned}$$

And since an implicator $\mathcal{I}$ is increasing in its second argument, we get:

$$\begin{aligned}\vec{E_{\mathcal{I}}}(\bigcap_{i=1}^{n} A_i, B)(y) &=_{HSV} \min_{x\in T_y(d_B)} \min_{i=1\ldots n} \mathcal{I}(B(x-y), A_i(x)) \\ &=_{HSV} \min_{i=1\ldots n} (\min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), A_1(x)), \ldots, \\ &\qquad \min_{x\in T_y(d_B)} \mathcal{I}(B(x-y), A_n(x))) \\ &=_{HSV} \bigcap_{i=1}^{n} \vec{E_{\mathcal{I}}}(A_i, B)(y), \quad \forall y \in \mathbb{R}^2.\end{aligned}$$

■

• Umbra approach

Definition 4.28. *Let A be a colour image and B a colour structuring element (both represented as $\mathbb{R}^2 - (HSV, \leq_{HSV})$ mappings). The* ***umbra 'colour' dilation*** *$\vec{D_u}(A, B)$ and the* ***umbra 'colour' erosion*** *$\vec{E_u}(A, B)$ are the colour images given by*

$$\begin{aligned}\vec{D}_u(A,B)(y) &\stackrel{def}{=}_{HSV} \max_{x\in T_y(d_B)} A(x) \oplus B(x-y) \qquad \textit{for } y \in \mathbb{R}^2, \\ \vec{E}_u(A,B)(y) &\stackrel{def}{=}_{HSV} \min_{x\in T_y(d_B)} A(x) \ominus B(x-y) \qquad \textit{for } y \in \mathbb{R}^2,\end{aligned}$$

where $\oplus$ and $\ominus$ are colour mix operators.

The question now is how to define the colour mix operators $\oplus$ and $\ominus$, both acting on colours in HSV and giving as result a colour in HSV. Because we want to obtain a complete lattice we require [56] that the dilation preserves the smallest element $\mathbf{0}$ and the erosion preserves the largest element $\mathbf{1}$ in HSV.

Definition 4.29. *Let c and c' be two colours in HSV. We define the colour mix operators $\oplus$ and $\ominus$ for c and c' as*

$$\begin{aligned}c \oplus c' &\stackrel{def}{=}_{HSV} \begin{cases} \mathbf{0}_{HSV} & \textit{if } c =_{HSV} \mathbf{0}_{HSV} \\ c +_{HSV} c' & \textit{otherwise} \end{cases}, \\ c \ominus c' &\stackrel{def}{=}_{HSV} \begin{cases} \mathbf{1}_{HSV} & \textit{if } c =_{HSV} \mathbf{1}_{HSV} \\ c -_{HSV} c' & \textit{otherwise} \end{cases}.\end{aligned}$$

Property 4.30.

$$\vec{D}_u(\overline{\mathbf{0}}, B) =_{HSV} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_u(\overline{\mathbf{1}}, B) =_{HSV} \overline{\mathbf{1}}.$$

Proof

$$\vec{D}_u(\overline{\mathbf{0}}, B)(y) \stackrel{def}{=}_{HSV} \max_{x\in T_y(d_B)} \overline{\mathbf{0}}(x) \oplus B(x-y) =_{HSV} \mathbf{0} =_{HSV} \overline{\mathbf{0}}(y), \ \ \forall y \in \mathbb{R}^2.$$

$$\vec{E}_u(\overline{\mathbf{1}}, B)(y) \stackrel{def}{=}_{HSV} \min_{x \in T_y(d_B)} \overline{\mathbf{1}}(x) \ominus B(x-y) =_{HSV} \mathbf{1} =_{HSV} \overline{\mathbf{1}}(y), \ \ \forall y \in \mathbb{R}^2.$$

■

Property 4.31 (Duality dilation-erosion).

$$\begin{aligned} \vec{D}_u(A,B) &=_{HSV} co(\vec{E}_u(co(A),B)) \\ \vec{E}_u(A,B) &=_{HSV} co(\vec{D}_u(co(A),B)). \end{aligned}$$

Proof
We first prove that $\vec{D}_u(A,B) =_{HSV} co(\vec{E}_u(co(A),B))$.

1) *If* $A \neq_{HSV} \overline{\mathbf{0}}$ *we get:*

$$\begin{aligned} co(\vec{E}_u(co(A),B))(y) &= \mathbf{1}_{HSV} - \vec{E}_u(\mathbf{1}_{HSV} - A, B)(y) \\ &= \mathbf{1}_{HSV} - \min_{x \in T_y(d_B)} (\mathbf{1}_{HSV} - A)(x) \ominus B(x-y) \\ &= \max_{x \in T_y(d_B)} \mathbf{1}_{HSV} - (\mathbf{1}_{HSV} - (A(x) +_{HSV} B(x-y))) \\ &= \max_{x \in T_y(d_B)} A(x) +_{HSV} B(x-y) \\ &= \vec{D}_u(A,B)(y), \ \forall y \in \mathbb{R}^2. \end{aligned}$$

2) *If* $A =_{HSV} \overline{\mathbf{0}}$ *we get:*

$$\begin{aligned} co(\vec{E}_u(co(A),B))(y) &=_{HSV} \mathbf{1}_{HSV} - \vec{E}_u(\mathbf{1}_{HSV} - \mathbf{0}_{HSV}, B)(y) \\ &=_{HSV} \mathbf{1}_{HSV} - \vec{E}_u(\mathbf{1}_{HSV}, B)(y) \\ &=_{HSV} \mathbf{1}_{HSV} - \min_{x \in T_y(d_B)} \mathbf{1}_{HSV}(x) \ominus B(x-y) \\ &=_{HSV} \mathbf{1}_{HSV} - \mathbf{1}_{HSV} \\ &=_{HSV} \mathbf{0}_{HSV} \\ &=_{HSV} \vec{D}_u(A,B)(y), \ \forall y \in \mathbb{R}^2. \end{aligned}$$

Combination of the cases above gives us that for every colour image A *and every colour structuring element* B *in HSV hold that* $\vec{D}_u(A,B) =_{HSV} co(\vec{E}_u(co(A),B))$.

We now prove that $\vec{E}_u(A,B) =_{HSV} co(\vec{D}_u(co(A),B))$.

1) *If* $A \neq_{HSV} \overline{\mathbf{1}}$ *we get:*

$$\begin{aligned} co(\vec{D}_u(co(A),B))(y) &= \mathbf{1}_{HSV} - \vec{D}_u(\mathbf{1}_{HSV} - A, B)(y) \\ &= \mathbf{1}_{HSV} - \max_{x \in T_y(d_B)} (\mathbf{1}_{HSV} - A)(x) \oplus B(x-y) \\ &= \min_{x \in T_y(d_B)} \mathbf{1}_{HSV} - ((\mathbf{1}_{HSV} - A)(x) \oplus B(x-y)) \\ &= \min_{x \in T_y(d_B)} A(x) -_{HSV} B(x-y) \\ &= \vec{E}_u(A,B)(y),\ \forall y \in \mathbb{R}^2. \end{aligned}$$

2) *If* $A =_{HSV} \overline{\mathbf{1}}$ *we get:*

$$\begin{aligned} co(\vec{D}_u(co(A),B))(y) &=_{HSV} \mathbf{1}_{HSV} - \vec{D}_u(\mathbf{1}_{HSV} - \mathbf{1}_{HSV}, B)(y) \\ &=_{HSV} \mathbf{1}_{HSV} - \vec{D}_u(\mathbf{0}_{HSV}, B)(y) \\ &=_{HSV} \mathbf{1}_{HSV} - \max_{x \in T_y(d_B)} \mathbf{0}_{HSV}(x) \oplus B(x-y) \\ &=_{HSV} \mathbf{1}_{HSV} - \mathbf{0}_{HSV} \\ &=_{HSV} \mathbf{1}_{HSV} \\ &=_{HSV} \vec{E}_u(A,B)(y),\ \forall y \in \mathbb{R}^2. \end{aligned}$$

Combination of the cases above gives us that for every colour image A *and every colour structuring element* B *in HSV hold that* $co(\vec{D}_u(co(A),B)) =_{HSV} \vec{E}_u(A,B)$.

■

Extension of greyscale morphology to colour morphology in L*a*b*

We extend the basic morphological operators dilation and erosion for greyscale images based on the threshold, umbra and fuzzy approach to colour images modelled in L*a*b*. For proofs we refer to the corresponding properties in HSV.

• Threshold approach

Let A be a colour image, represented as a $\mathbb{R}^2 - (L^*a^*b^*, \leq_{L^*a^*b^*})$ mapping, and B a binary structuring element ($\subseteq \mathbb{R}^2$). The support of A is defined as

$$\{x \in \mathbb{R}^2 \mid A(x) >_{L^*a^*b^*} \mathbf{0}\}.$$

Definition 4.32. *Let* A *be a colour image and* B *a binary structuring element. The* ***threshold 'colour' dilation*** $\vec{D}_t(A,B)$ *and the* ***threshold 'colour' erosion*** $\vec{E}_t(A,B)$

are the colour images given by

$$\vec{D}_t(A,B)(y) \stackrel{def}{=}_{L^*a^*b^*} \max_{x\in T_y(B)} A(x) \qquad \textit{for } y\in\mathbb{R}^2,$$

$$\vec{E}_t(A,B)(y) \stackrel{def}{=}_{L^*a^*b^*} \min_{x\in T_y(B)} A(x) \qquad \textit{for } y\in\mathbb{R}^2.$$

Property 4.33.

$$\vec{D}_t(\overline{\mathbf{0}},B) =_{L^*a^*b^*} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(\overline{\mathbf{1}},B) =_{L^*a^*b^*} \overline{\mathbf{1}}$$
$$\vec{D}_t(A,\emptyset) =_{L^*a^*b^*} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(A,\emptyset) =_{L^*a^*b^*} \overline{\mathbf{1}}.$$

Property 4.34 (Duality dilation-erosion).

$$\vec{D}_t(A,B) \quad =_{L^*a^*b^*} \quad co(\vec{E}_t(co(A),B))$$
$$\vec{E}_t(A,B) \quad =_{L^*a^*b^*} \quad co(\vec{D}_t(co(A),B)).$$

Property 4.35 (Monotonicity). *If A and B are two colour images, and C and C' are two binary structuring elements, then it holds that*

$$\begin{aligned} A \subseteq_{L^*a^*b^*} B \quad &\Rightarrow \quad \vec{D}_t(A,C) \subseteq_{L^*a^*b^*} \vec{D}_t(B,C) \textit{ and} \\ &\qquad\quad \vec{E}_t(A,C) \subseteq_{L^*a^*b^*} \vec{E}_t(B,C) \\ C \subseteq C' \quad &\Rightarrow \quad \vec{D}_t(A,C) \subseteq_{L^*a^*b^*} \vec{D}_t(A,C') \textit{ and} \\ &\qquad\quad \vec{E}_t(A,C) \supseteq_{L^*a^*b^*} \vec{E}_t(A,C'). \end{aligned}$$

Property 4.36 (Inclusion).

$$\vec{E}_t(A,B) \subseteq_{L^*a^*b^*} \vec{D}_t(A,B).$$

Property 4.37.

$$\mathbf{0}\in B \quad \Rightarrow \quad A \subseteq_{L^*a^*b^*} \vec{D}_t(A,B) \textit{ and } \vec{E}_t(A,B) \subseteq_{L^*a^*b^*} A.$$

Property 4.38 (Interaction with intersection and union). *Consider a family $(A_i)_{i=1}^n$ of colour images and a family $(B_i)_{i=1}^n$ of binary structuring elements.*
For the t-'colour' dilation it holds that

$$\vec{D}_t(\bigcap_{i=1}^n A_i,B) \quad \subseteq_{L^*a^*b^*} \quad \bigcap_{i=1}^n \vec{D}_t(A_i,B)$$
$$\vec{D}_t(A,\bigcap_{i=1}^n B_i) \quad \subseteq_{L^*a^*b^*} \quad \bigcap_{i=1}^n \vec{D}_t(A,B_i);$$
$$\vec{D}_t(\bigcup_{i=1}^n A_i,B) \quad =_{L^*a^*b^*} \quad \bigcup_{i=1}^n \vec{D}_t(A_i,B)$$
$$\vec{D}_t(A,\bigcup_{i=1}^n B_i) \quad =_{L^*a^*b^*} \quad \bigcup_{i=1}^n \vec{D}_t(A,B_i).$$

For the t-'colour' erosion it holds that

$$\begin{aligned}
\vec{E}_t(\bigcap_{i=1}^{n} A_i, B) &=_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{E}_t(A_i, B) \\
\vec{E}_t(A, \bigcap_{i=1}^{n} B_i) &\supseteq_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{E}_t(A, B_i); \\
\vec{E}_t(\bigcup_{i=1}^{n} A_i, B) &\supseteq_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{E}_t(A_i, B) \\
\vec{E}_t(A, \bigcup_{i=1}^{n} B_i) &=_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{E}_t(A, B_i).
\end{aligned}$$

Property 4.39. *Let A be a colour image and B and C two binary structuring elements, then*

$$\begin{aligned}
\vec{D}_t(\vec{D}_t(A, B), C) &=_{L^*a^*b^*} \vec{D}_t(\vec{D}_t(A, C), B) \\
\vec{E}_t(\vec{E}_t(A, B), C) &=_{L^*a^*b^*} \vec{E}_t(\vec{E}_t(A, C), B).
\end{aligned}$$

• Fuzzy approach

The support d_A of a colour image A in L*a*b* is defined as the set

$$d_A = \{x \in \mathbb{R}^2 \mid A(x) >_{L^*a^*b^*} \mathbf{0}\}.$$

Definition 4.40. *Let A be a colour image and B a colour structuring element (both seen as $(L^*a^*b^*, \leq_{L^*a^*b^*})$-fuzzy sets), $\mathcal{C}$ a conjunctor on $(L^*a^*b^*, \leq_{L^*a^*b^*})$ and $\mathcal{I}$ an implicator on $(L^*a^*b^*, \leq_{L^*a^*b^*})$. The **fuzzy 'colour' dilation** $\vec{D}_{\mathcal{C}}(A, B)$ and the **fuzzy 'colour' erosion** $\vec{E}_{\mathcal{I}}(A, B)$ are the $(L^*a^*b^*, \leq_{L^*a^*b^*})$-fuzzy sets defined as*

$$\begin{aligned}
\vec{D}_{\mathcal{C}}(A, B)(y) &\stackrel{def}{=}_{L^*a^*b^*} \max_{x \in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) \qquad \text{for } y \in \mathbb{R}^2, \\
\vec{E}_{\mathcal{I}}(A, B)(y) &\stackrel{def}{=}_{L^*a^*b^*} \min_{x \in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) \qquad \text{for } y \in \mathbb{R}^2.
\end{aligned}$$

Notice that we can write

$$\begin{aligned}
\max_{x \in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) &=_{L^*a^*b^*} \max_{x \in \mathbb{R}^2} \mathcal{C}(R_B(x, y), A(x)), \\
\min_{x \in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) &=_{L^*a^*b^*} \min_{x \in \mathbb{R}^2} \mathcal{I}(R_B(x, y), A(x)),
\end{aligned}$$

for all y in $\mathbb{R}^2$, whereby

$$R_B : \mathbb{R}^2 \times \mathbb{R}^2 \to (L^*a^*b^*, \leq_{L^*a^*b^*}),$$

$R_B = B \circ V$ with V defined as $V(x,y) = x - y = (x_1, x_2) - (y_1, y_2) = (x_1 - y_1, x_2 - y_2),\ \forall (x,y) \in (\mathbb{R}^2)^2$ so that $R_B(x,y) = B(x-y)$.

With every structuring element B we can associate a $(L^*a^*b^*, \leq_{L^*a^*b^*})$-fuzzy relation R_B on $\mathbb{R}^2$.

Property 4.41. *[12] Let $\mathcal{T}$ be a t-norm and $\mathcal{I}$ be an implicator on $(L^*a^*b^*, \leq_{L^*a^*b^*})$, then it holds:*

$$\vec{D}_{\mathcal{T}}(\overline{\mathbf{0}}, B) =_{L^*a^*b^*} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{1}}, B) =_{L^*a^*b^*} \overline{\mathbf{1}}.$$

If $B(\mathbf{0}) = \mathbf{1}$, then it holds:

$$\vec{D}_{\mathcal{T}}(\overline{\mathbf{1}}, B) =_{L^*a^*b^*} \overline{\mathbf{1}} \quad \textit{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{0}}, B) =_{L^*a^*b^*} \overline{\mathbf{0}}.$$

Property 4.42 (Duality dilation-erosion). *[12] Let $\mathcal{T}$ be a t-norm on $L^*a^*b^*$, $\mathcal{N}$ an involutive negator and $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ the corresponding $\mathcal{S}$-implicator on $L^*a^*b^*$. For every colour image A and colour structuring element B we have*

$$\begin{aligned} co_{\mathcal{N}}\vec{D}_{\mathcal{T}}(A,B) &=_{L^*a^*b^*} \vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(co_{\mathcal{N}}A, B) \\ \vec{D}_{\mathcal{T}}(co_{\mathcal{N}}A, B) &=_{L^*a^*b^*} co_{\mathcal{N}}(\vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(A,B)). \end{aligned}$$

Property 4.43 (Monotonicity). *[Generalisation of [12]] If A and B are two colour images, C and C' two colour structuring elements, $\mathcal{C}_1$ and $\mathcal{C}_2$ two conjunctors and $\mathcal{I}_1$ and $\mathcal{I}_2$ two implicators on $L^*a^*b^*$, then it holds that*

$$\begin{aligned} A \subseteq_{L^*a^*b^*} B \quad &\Rightarrow \quad \vec{D}_{\mathcal{C}}(A,C) \subseteq_{L^*a^*b^*} \vec{D}_{\mathcal{C}}(B,C) \textit{ and} \\ &\qquad\ \vec{E}_{\mathcal{I}}(A,C) \subseteq_{L^*a^*b^*} \vec{E}_{\mathcal{I}}(B,C) \\ C \subseteq_{L^*a^*b^*} C' \quad &\Rightarrow \quad \vec{D}_{\mathcal{C}}(A,C) \subseteq_{L^*a^*b^*} \vec{D}_{\mathcal{C}}(A,C') \textit{ and} \\ &\qquad\ \vec{E}_{\mathcal{I}}(A,C) \supseteq_{L^*a^*b^*} \vec{E}_{\mathcal{I}}(A,C') \\ \mathcal{C}_1 \subseteq_{L^*a^*b^*} \mathcal{C}_2 \quad &\Rightarrow \quad \vec{D}_{\mathcal{C}_1}(A,C) \subseteq_{L^*a^*b^*} \vec{D}_{\mathcal{C}_2}(A,C) \\ \mathcal{I}_1 \subseteq_{L^*a^*b^*} \mathcal{I}_2 \quad &\Rightarrow \quad \vec{E}_{\mathcal{I}_1}(A,C) \subseteq_{L^*a^*b^*} \vec{E}_{\mathcal{I}_2}(A,C). \end{aligned}$$

Property 4.44 (Inclusion). *[Generalisation of [12]] Let $\mathcal{C}$ be a seminorm on L*a*b* and $\mathcal{I}$ an edge-implicator on L*a*b*. Consider a colour image A and a 'normalized' colour structuring element B, that is, $(\forall y \in \mathbb{R}^2)(\exists z \in \mathbb{R}^2)(B(z-y) =_{L^*a^*b^*} \mathbf{1})$. It holds that*

$$\vec{E}_{\mathcal{I}}(A,B) \subseteq_{L^*a^*b^*} \vec{D}_{\mathcal{C}}(A,B).$$

Property 4.45. *[Generalisation of [12]] Let $\mathcal{C}$ be a seminorm on L*a*b* and $\mathcal{I}$ an edge-implicator on L*a*b*. For every colour image A and every colour structuring element B, it holds that*

$$B(\mathbf{0}) =_{L^*a^*b^*} \mathbf{1} \quad \Rightarrow \quad A \subseteq_{L^*a^*b^*} \vec{D}_{\mathcal{C}}(A,B) \textit{ and } \vec{E}_{\mathcal{I}}(A,B) \subseteq_{L^*a^*b^*} A.$$

Property 4.46 (Interaction with intersection and union). *[Generalisation of [12]] Consider a family $(A_i)_{i=1}^n$ of colour images and a family $(B_i)_{i=1}^n$ of colour structuring elements. For the $\mathcal{C}$-'colour' dilation it holds that*

$$\begin{aligned}
\vec{D}_{\mathcal{C}}(\bigcap_{i=1}^{n} A_i, B) &\subseteq_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{D}_{\mathcal{C}}(A_i, B)\\
\vec{D}_{\mathcal{C}}(A, \bigcap_{i=1}^{n} B_i) &\subseteq_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{D}_{\mathcal{C}}(A, B_i);\\
\vec{D}_{\mathcal{C}}(\bigcup_{i=1}^{n} A_i, B) &=_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{D}_{\mathcal{C}}(A_i, B)\\
\vec{D}_{\mathcal{C}}(A, \bigcup_{i=1}^{n} B_i) &=_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{D}_{\mathcal{C}}(A, B_i).
\end{aligned}$$

For the $\mathcal{I}$-'colour' erosion it holds that

$$\begin{aligned}
\vec{E}_{\mathcal{I}}(\bigcap_{i=1}^{n} A_i, B) &=_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{E}_{\mathcal{I}}(A_i, B)\\
\vec{E}_{\mathcal{I}}(A, \bigcap_{i=1}^{n} B_i) &\supseteq_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{E}_{\mathcal{I}}(A, B_i);\\
\vec{E}_{\mathcal{I}}(\bigcup_{i=1}^{n} A_i, B) &\supseteq_{L^*a^*b^*} \bigcup_{i=1}^{n} \vec{E}_{\mathcal{I}}(A_i, B)\\
\vec{E}_{\mathcal{I}}(A, \bigcup_{i=1}^{n} B_i) &=_{L^*a^*b^*} \bigcap_{i=1}^{n} \vec{E}_{\mathcal{I}}(A, B_i).
\end{aligned}$$

• Umbra approach

Definition 4.47. *Let A be a colour image and B a colour structuring element (both represented as $\mathbb{R}^2 - (L^*a^*b^*, \leq_{L^*a^*b^*})$ mappings). The **umbra 'colour' dilation** $\vec{D}_u(A, B)$ and the **umbra 'colour' erosion** $\vec{E}_u(A, B)$ are the colour images given by*

$$\begin{aligned}
\vec{D}_u(A, B)(y) &\stackrel{def}{=}_{L^*a^*b^*} \max_{x \in T_y(d_B)} A(x) \oplus B(x - y) \qquad \text{for } y \in \mathbb{R}^2,\\
\vec{E}_u(A, B)(y) &\stackrel{def}{=}_{L^*a^*b^*} \min_{x \in T_y(d_B)} A(x) \ominus B(x - y) \qquad \text{for } y \in \mathbb{R}^2,
\end{aligned}$$

where $\oplus$ and $\ominus$ are colour mix operators.

Definition 4.48. *Let c and c' be two colours in L*a*b*. We define the colour mix*

operators $\oplus$ *and* $\ominus$ *for* c *and* c' *as*

$$c \oplus c' =_{L^*a^*b^*} \begin{cases} \mathbf{0}_{L^*a^*b^*} & \text{if } c =_{L^*a^*b^*} \mathbf{0}_{L^*a^*b^*} \\ c +_{L^*a^*b^*} c' & \text{otherwise} \end{cases},$$

$$c \ominus c' =_{L^*a^*b^*} \begin{cases} \mathbf{1}_{L^*a^*b^*} & \text{if } c =_{L^*a^*b^*} \mathbf{1}_{L^*a^*b^*} \\ c -_{L^*a^*b^*} c' & \text{otherwise} \end{cases}.$$

Property 4.49.

$$\vec{D}_u(\overline{\mathbf{0}}, B) =_{L^*a^*b^*} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_u(\overline{\mathbf{1}}, B) =_{L^*a^*b^*} \overline{\mathbf{1}}.$$

Property 4.50 (Duality dilation-erosion).

$$\begin{aligned} \vec{D}_u(A, B) &=_{L^*a^*b^*} co(\vec{E}_u(co(A), B)) \\ \vec{E}_u(A, B) &=_{L^*a^*b^*} co(\vec{D}_u(co(A), B)). \end{aligned}$$

4.3.6 Experimental Results

Consider now a colour image C, modelled in the HSV or L*a*b* colour model, and a one- or three-dimensional structuring element B_{HSV} or $B_{L^*a^*b^*}$. For the extension of the greyscale morphological operators to morphological operators acting on colour images we get

1. The t-'colour' morphological operators (threshold approach):
   We calculate the maximum and minimum of the set of colours of the image C contained in a $m \times m$ window (structuring element) around a chosen central colour pixel. The t-'colour' dilation and t-'colour' erosion are the original colours of the pixels where this maximum, respectively minimum, is obtained.

2. The u-'colour' morphological operators (umbra approach):
   First we mix the colours (addition for the u-dilation, subtraction for the u-erosion) of the original image C with the colours of our chosen structuring element B in the considered window. Secondly we determine the maximum and minimum of this new set of colours for the u-'colour' dilation and u-'colour' erosion. The u-'colour' dilation and u-'colour' erosion are the new colours of the pixels where this maximum, respectively minimum, is obtained. But we can also look for the positions in C where this maximum or minimum is reached. And the u-'colour' dilation and u-'colour' erosion can then be given by the original colours (in the original image C) of these pixels.

3. The fuzzy 'colour' morphological operators (fuzzy logic approach):
   Again, we have to determine the maximum and minimum of a (new) set of colours, possibly after adding, subtracting or multiplying original colours of C with colours of the structuring element B.

Finally in our experimental results (figure 4.11 to 4.24) we have compared our new approach with the component-based approach. And what is more, we have compared our u-morphological colour operators in HSV with the u-morphological colour operators proposed in [39] and our t-morphological colour operators in L*a*b* with the t-morphological colour operators proposed in [4]. We have used different test images in our experiments (the well-known Tulips, Trees and Lena images), shown in figure 4.10. Because the dilation is a supremum operator, this operator will suppress dark colours and intensify light colours: objects/areas in the image that have a dark colour become smaller while objects/areas that have a light colour become larger. The erosion on the other hand is an infimum operator so that light colours are suppressed and dark colours intensified. The choice of the structuring element has of course a great influence on the result and will obviously depend on the application. As 'binary' structuring elements we have used

$$B'(i,j,1) = B'(i,j,2) = B'(i,j,3) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \underline{1} & 1 \\ 1 & 1 & 1 \end{pmatrix},\ 1 \leq i,j, \leq 3,$$

or

$$B' * (i,j,1) = B' * (i,j,2) = B' * (i,j,3) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \underline{0} & 0 \\ 0 & 0 & 0 \end{pmatrix},\ 1 \leq i,j, \leq 3,$$

or

$$B''*(i,j,1) = B''*(i,j,2) = B''*(i,j,3) = \begin{pmatrix} 0.55 & 0.55 & 0.55 \\ 0.55 & \underline{0.55} & 0.55 \\ 0.55 & 0.55 & 0.55 \end{pmatrix},\ 1 \leq i,j, \leq 3,$$

and as greyscale structuring element

$$B''_{RGB}(i,j,1) = B''_{RGB}(i,j,2) = B''_{RGB}(i,j,3) = \begin{pmatrix} 0 & 255 & 0 \\ 255 & \underline{255} & 255 \\ 0 & 255 & 0 \end{pmatrix},$$

or

$$B^{Wh}_{RGB}(i,j,1) = B^{Wh}_{RGB}(i,j,2) = B^{Wh}_{RGB}(i,j,3) = \begin{pmatrix} 255 & 255 & 255 \\ 255 & \underline{255} & 255 \\ 255 & 255 & 255 \end{pmatrix},$$

$1 \leq i,j, \leq 3$, where the underlined element corresponds to the origin of coordinates. Notice that since in both the HSV and L*a*b* colour model we can separate intensity from chrominance (= hue and saturation or chroma), we will obtain the best results for the component-based approach by applying the greyscale morphological operators on the intensity component only. Then we add the 'new' intensity component to the original chrominance components to get again a colour image in the HSV or L*a*b* colour model.

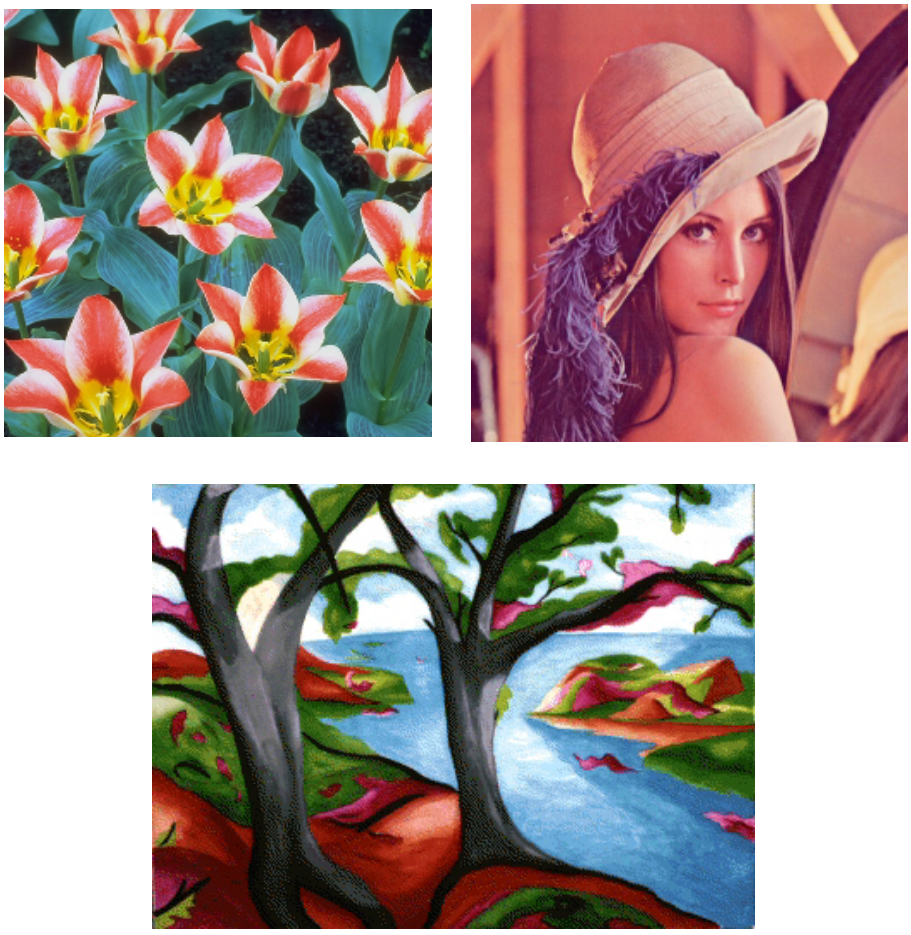


Figure 4.10: At the top, from left to right: the original Tulips and Lena image, and at the bottom, the original Trees image.

In figures 4.11 and 4.12 the t-colour dilation and t-colour erosion in HSV and L*a*b* of the component-based approach and the proposed method are shown. Pay attention to the edge of the tulips leaves. New colours appear with the component-based approach, while with our approach the colours are preserved.

Figure 4.11: T-morphological operators in HSV: at the top: the original image C, left column: the t-dilation $D_t(C, B')$ and right column: the t-erosion $E_t(C, B')$: from top to bottom: the component-based approach and our new vector-based approach.

Figure 4.12: T-morphological operators in L*a*b*: at the top: the original image C, left column: the t-dilation $D_t(C, B')$ and right column: the t-erosion $E_t(C, B')$: from top to bottom: the component-based approach and our new vector-based approach.

We have compared the results for the t-morphological colour operators in L*a*b* of our method with these of the state-of-the-art method [4] in figure 4.13. The difference between the two approaches is noticeable at the edge of the tulips leaves, but the results are quite similar.

Figure 4.13: T-morphological colour operators in L*a*b* by structuring element B': at the top: the t-dilation and at the bottom: the t-erosion, using our RGB ordering (left) and the reduced ordering, based on the distance to white, completed by the conditional ordering L* $\rightarrow$ a* $\rightarrow$ b* (right).

Figures 4.14 and 4.15 illustrate the fuzzy morphological colour operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s})$ in HSV and L*a*b* obtained by the component-based approach and the proposed approach. Look at the edge of the hat. With the component-based approach new colours are introduced, whereas with our method no colours that are not present in the original colour image appear in the results.

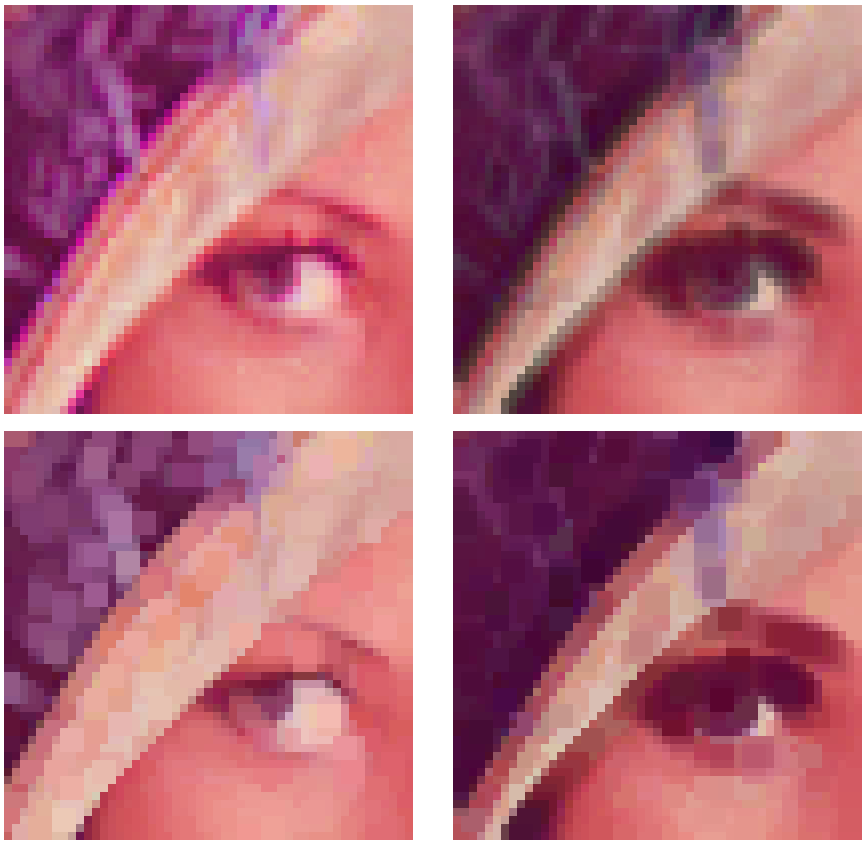


Figure 4.14: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s})$ in HSV: left column: the dilations $D_{\mathcal{T}_{\min}}(C, B^{Wh})$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}}(C, B^{Wh})$: from top to bottom: the component-based approach and our new vector-based approach.

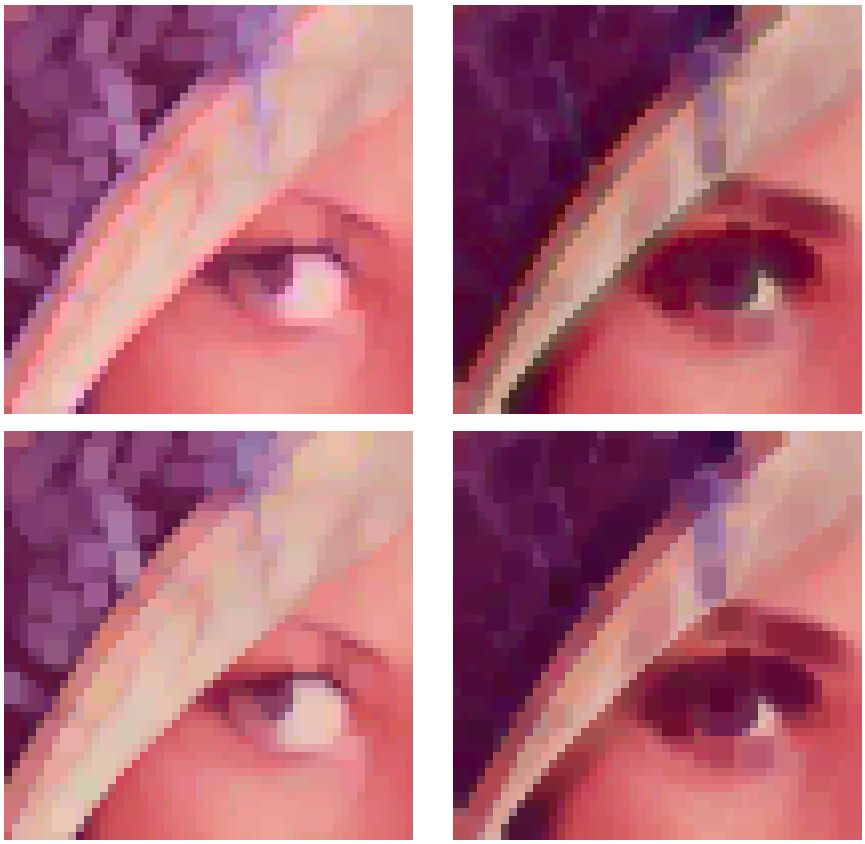


Figure 4.15: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s})$ in L*a*b*: left column: the dilations $D_{\mathcal{T}_{\min}}(C, B^{Wh})$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_{\min}}, \mathcal{N}_s}(C, B^{Wh})$: from top to bottom: the component-based approach and our new approach.

The fuzzy morphological colour dilation and colour erosion for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in HSV and L*a*b* for the component-based approach and the proposed approach are shown in figures 4.16 and 4.17. Pay attention to the edge of the trees, where the colours are not preserved with the component-based approach but no new colours appear with our method.

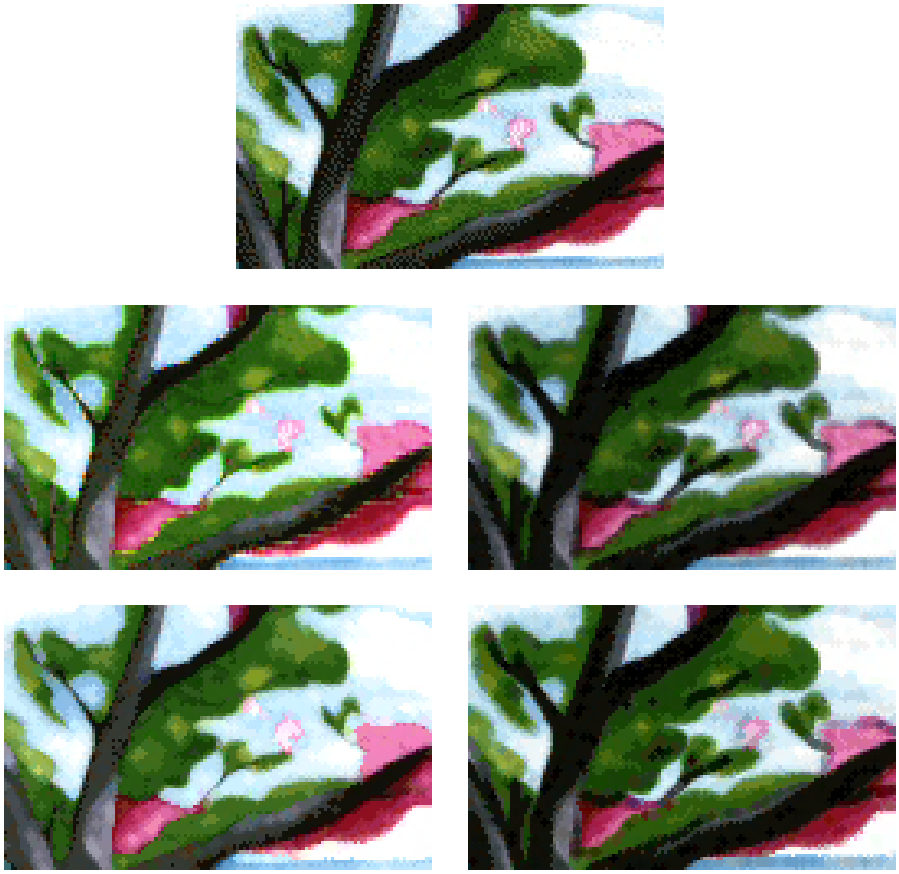


Figure 4.16: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in HSV: at the top: the original image C, left column: the dilations $D_{\mathcal{T}_*}(C, B'')$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$: from top to bottom: the component-based approach and our new vector-based approach.

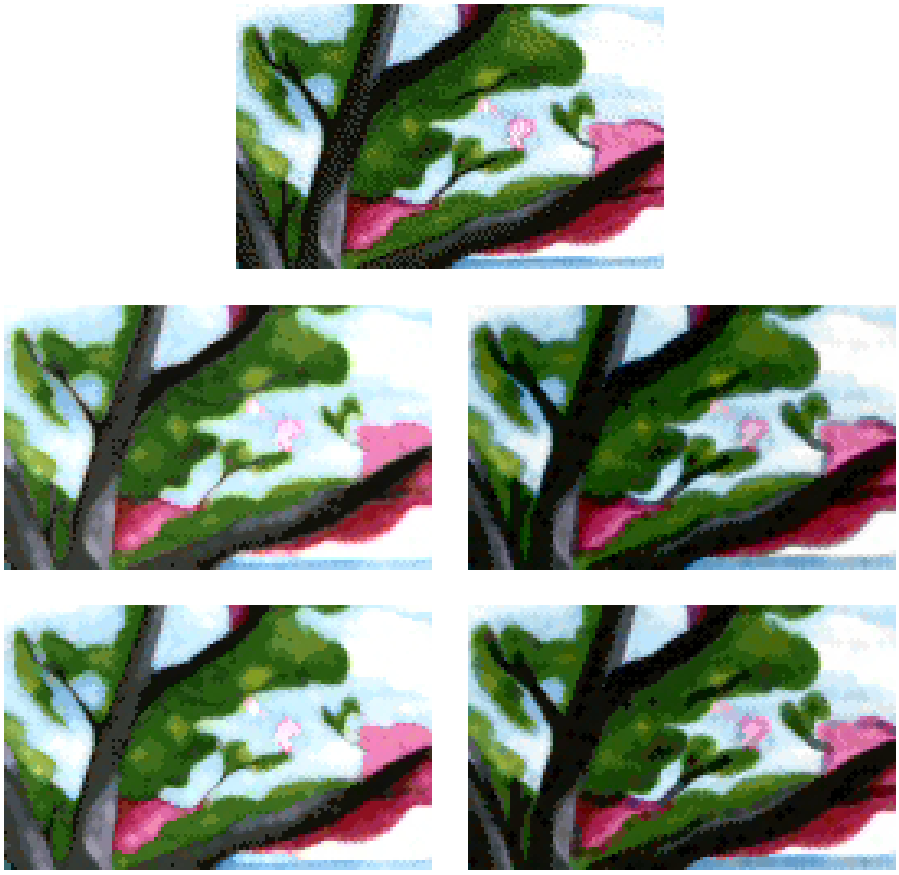


Figure 4.17: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in L*a*b*: at the top: the original image C, left column: the dilations $D_{\mathcal{T}_*}(C, B'')$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$: from top to bottom: the component-based approach and our new vector-based approach.

Note that for the fuzzy morphological operators with conjunctor-implicator pair $(\mathcal{C},\mathcal{I}) = (\mathcal{T}_*,\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s})$ we have used the structuring element B''. If we choose as structuring element for example

$$B'''_{RGB}(i,j,1) = B'''_{RGB}(i,j,2) = B'''_{RGB}(i,j,3) = \frac{1}{255}\begin{pmatrix} 155 & 235 & 155 \\ 235 & \underline{255} & 235 \\ 155 & 235 & 155 \end{pmatrix},$$

$1 \leq i,j, \leq 3$, new unwanted colours that are not present in the original image can appear in the image after applying the morphological operators, as shown along the edges of the tulips leaves in figure 4.18.

Figure 4.18: Fuzzy morphological operators for $(\mathcal{C},\mathcal{I}) = (\mathcal{T}_*,\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s})$ in HSV: at the top: the original image C, the fuzzy dilation $D_{\mathcal{T}_*}(C,B''')$ (left) and the fuzzy erosion $E_{\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s}}(C,B''')$ obtained by our new approach.

To solve this problem we better first multiply or subtract colours of the original image C with colours of the chosen structuring element B''' and then determine the maximum and minimum of this new set of colours. Next we look at the positions in C where this maximum or minimum is reached. The fuzzy dilation and fuzzy erosion for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ are then given by the original colours (in the original image C) of these pixels.

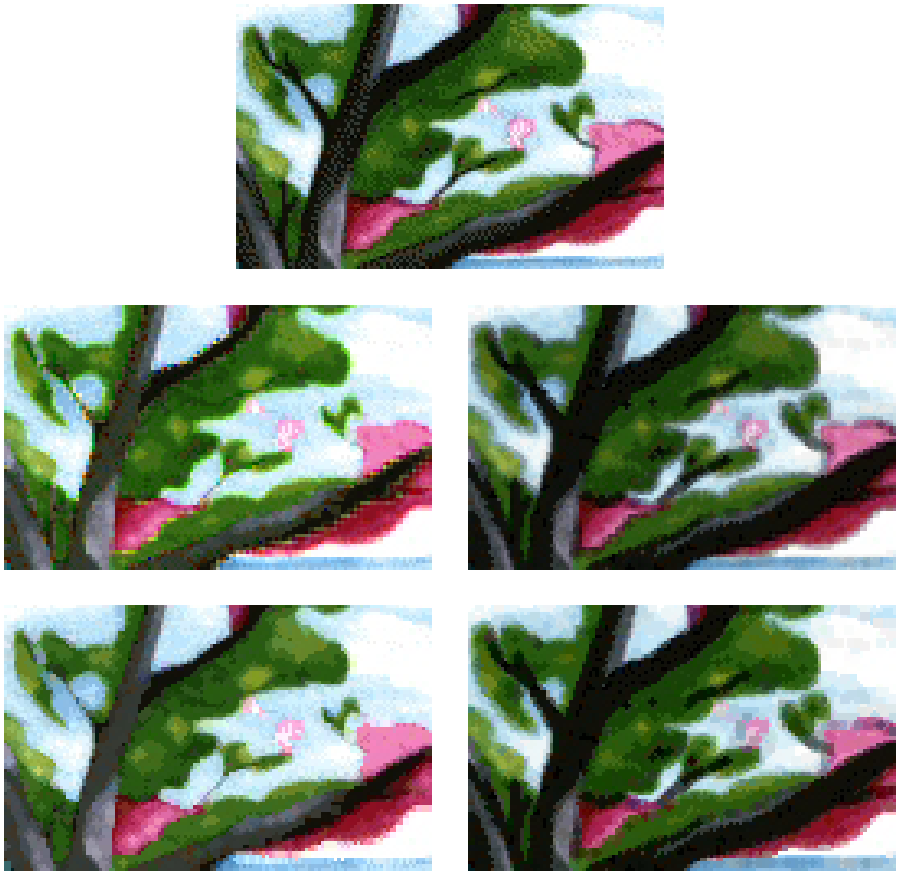


Figure 4.19: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in HSV: at the top: the original image C, left column: the dilations $D_{\mathcal{T}_*}(C, B''')$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B''')$: from top to bottom: the component-based approach and our new vector-based approach.

Figure 4.19 and 4.20 show the fuzzy colour dilation and erosion for the conjunctor-implicator pair $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s})$ in HSV and L*a*b* for the component-based and the proposed approach. Along the edges of the trees new colours appear with the component-based approach, which is not the case with the new method.

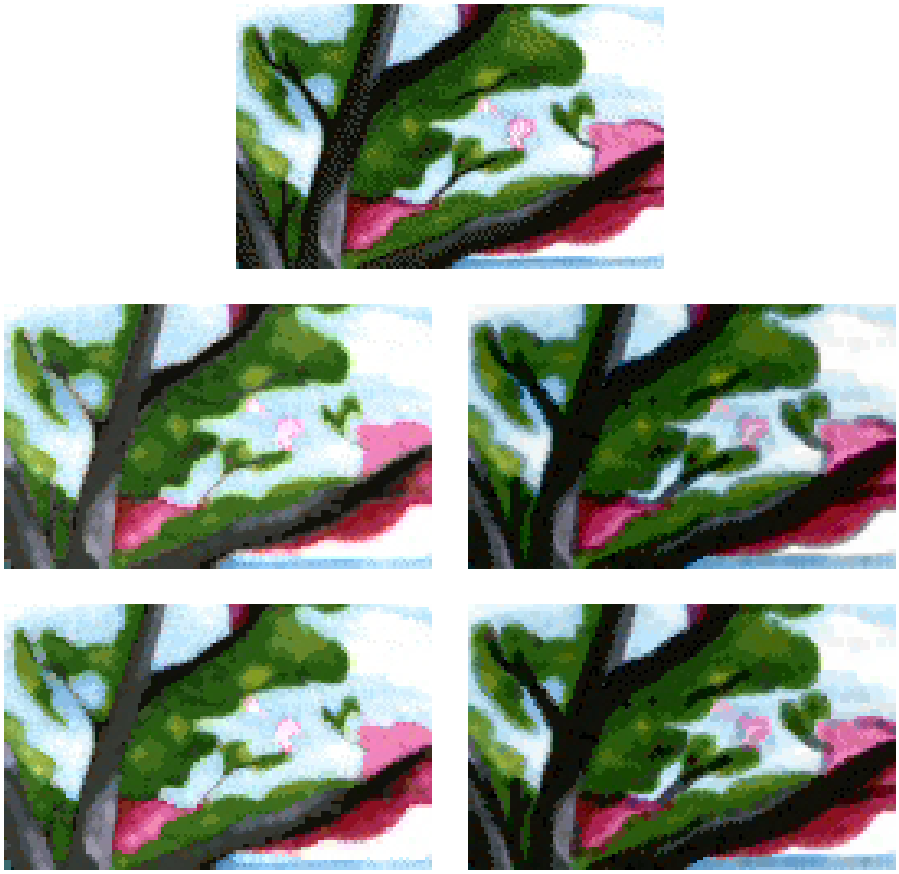


Figure 4.20: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s})$ in L*a*b*: at the top: the original image C, left column: the dilations $D_{\mathcal{T}_*}(C, B''')$ and right column: the erosions $E_{\mathcal{I}_{\mathcal{T}_*,\mathcal{N}_s}}(C, B''')$: from top to bottom: the component-based approach and our new vector-based approach.

The u-colour dilation and erosion using the new vector-based approach in the HSV and L*a*b* colour model is given in figure 4.21 and 4.22 respectively. The results of the proposed approach when replacing the new colours by the original colours of the corresponding pixels are also shown.

Figure 4.21: U-morphological operators in HSV: left column: the u-dilation $D_u(C, B^{Wh})$ and right column: the u-erosion $E_u(C, B^{Wh})$: from top to bottom: our new vector-based approach and the result of our approach when replacing the new colours by the original colours of the corresponding pixel positions.

Figure 4.22: U-morphological operators in L*a*b*: at the top: the original image C, left column: the u-dilation $D_u(C, B^{Wh})$ and right column: the u-erosion $E_u(C, B^{Wh})$: from top to bottom: our new vector-based approach and the result of our approach when replacing the new colours by the original colours of the corresponding pixel positions.

In figure 4.23 and 4.24 the results of the u-morphological colour operators in HSV by the proposed method and the state-of-the-art method described in [39] are illustrated. Figure 4.23 shows that the colours obtained by our approach are natural in comparison with the original colours of the original image, while artificial colours appear in the images using the other approach. The results of both approaches when replacing the new colours by the original colours of the right pixel positions, as shown in figure 4.24, are very similar.

Figure 4.23: U-morphological colour operators in HSV: at the top: the u-'colour' dilation based on our approach by structuring element B^{Wh} and based on the approach [39] by structuring element $B''*$, and at the bottom: the u-'colour'erosion based on our approach by structuring element B^{Wh} and based on the approach [39] by structuring element $B''*$.

Figure 4.24: U-morphological colour operators in HSV: at the top: the u-'colour' dilation based on our approach by structuring element B^{Wh} and based on the approach [39] by structuring element $B'*$, both results when replacing the new colours by the original colours of the corresponding pixel positions, and at the bottom: the u-'colour'erosion based on our approach by structuring element B^{Wh} and based on the approach [39] by structuring element $B'*$, both results when replacing the new colours by the original colours of the corresponding pixel positions.

4.3.7 New RGB Colour Ordering Compatible with the Complement co

Now we introduce a new approach for the ordering of colours in RGB. We first explain our idea and define a new ordering $\leq_{RGB}$ compatible with the complement co (we need this property for our morphological interpolation method to magnify images (see chapter 4)) so that $(RGB, \leq_{RGB})$ becomes a lattice.

Construction of the new RGB colour vector ordering $\leq_{RGB}$

Our idea still is to rank colours in RGB from 'dark' colours (close to black) to 'light' colours (close to white). When we look at the distance of colours in RGB to black and white (see section 4.3.2), we do not get an order relation. So we got the idea to consider the centre $(1/2, 1/2, 1/2)$ of the RGB cube (as the middle of the black and white top) and to determine from this point if colours are lying close to black or close to white.

In the RGB cube we consider a plane V through the centre $m = (1/2, 1/2, 1/2)$ perpendicular to the line l determined by the two points Bl $(0, 0, 0)$ and Wh $(1, 1, 1)$. The line l is determined as intersection of the two planes $r = g$ and $g = b$. In figure 4.25 you see the line l and the plane V in the RGB cube. The equation of the plane V with non-zero normal vector $(1, 1, 1)$ through $(1/2, 1/2, 1/2)$ is

$$V: \; r + g + b - \frac{3}{2} = 0.$$

For every colour $c = (r_c, g_c, b_c)$ in RGB we now look if c lies 'below' or 'above' the plane V with respect to its normal vector $(1, 1, 1)$. So we have to work out the "distance" from c to V as follows

$$D_{(c,V)} = \frac{(r_c + g_c + b_c - 3/2)}{\sqrt{3}}.$$

If $D_{(c,V)} > 0$, then c is on the same side of the plane as the normal vector $(1, 1, 1)$; if $D_{(c,V)} < 0$, then c is on the opposite side; and if $D_{(c,V)} = 0$, then c lies in V. This way we can distinguish between 'dark' colours lying close to black ($D_{(c,V)} < 0$) and 'light' colours lying close to white ($D_{(c,V)} > 0$), where we will rank 'dark' colours lower than 'light' colours. The plane V 'divides' the RGB cube into two similar parts.

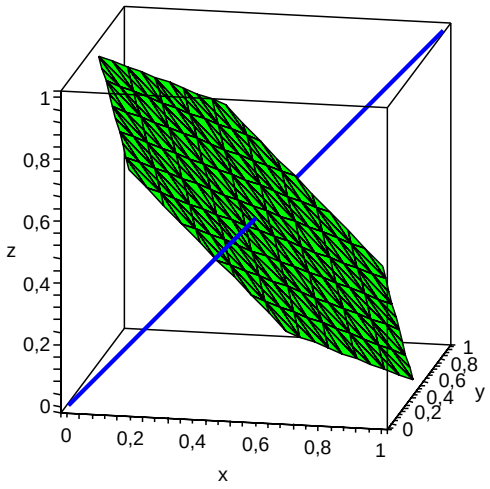


Figure 4.25: The RGB colour cube with the plane V (in green) and the line l (in blue).

Let $c = (r_c, g_c, b_c)$ and $c' = (r_{c'}, g_{c'}, b_{c'})$ be two colours in RGB. We will consider the following cases for the ranking of c and c':

1^{st} case: The two colours c and c' are not lying on the same side w.r.t. V

1.1. c lies under V and c' lies above V, i.e., $D_{(c,V)} < 0$ and $D_{(c',V)} > 0$

1.2. c lies in V and c' lies above V, i.e., $D_{(c,V)} = 0$ and $D_{(c',V)} > 0$

1.3. c lies under V and c' lies in V, i.e., $D_{(c,V)} < 0$ and $D_{(c',V)} = 0$.

In these three subcases c is ranked lower than c', i.e., $c <_{RGB} c'$.

2^{nd} case: The two colours c and c' are lying on the same side w.r.t. V

2.1. c and c' are lying above V, i.e., $D_{(c,V)} > 0$ and $D_{(c',V)} > 0$; c and c' lie both close to white

All colours above V are considered to be 'light' colours lying close to white. Because we want to determine which of the two colours c and c' is the 'lightest' colour, we slice the cube by looking at the distance to V:

$$D(c, V) = |r_c + g_c + b_c - 3/2| / \sqrt{3} = \left|D_{(c,V)}\right|$$

$$D(c', V) = |r_{c'} + g_{c'} + b_{c'} - 3/2| / \sqrt{3} = \left|D_{(c',V)}\right|$$

where the colours c and c' are then ordered accordingly to their distance with respect to V.

2.1.1. $D(c, V) < D(c', V)$: c is a 'darker' colour than c' so that we rank c lower than c', i.e., $c <_{RGB} c'$.

2.1.2. $D(c, V) > D(c', V)$: c is a 'lighter' colour than c' so that we rank c higher than c', i.e., $c >_{RGB} c'$.

2.1.3. $D(c, V) = D(c', V) \rightarrow$ the distance to m is taken into account:

If ($D_{(c,V)} > 0$ and $D_{(c',V)} > 0$) and ($D(c, V) = D(c', V)$), thus $D_{(c,V)} = D_{(c',V)} > 0$, that is, if the two colours c and c' lie on the same plane $W_{c,c'}$ parallel to V at distance $D(c, V) = D(c', V)$, then we determine the distance from c and c' to the centre m of the cube as

$$D(c, m) = \sqrt{(r_c - 1/2)^2 + (g_c - 1/2)^2 + (b_c - 1/2)^2},$$

$$D(c', m) = \sqrt{(r_{c'} - 1/2)^2 + (g_{c'} - 1/2)^2 + (b_{c'} - 1/2)^2}.$$

We sort the colours w.r.t. their distance to m, that is,

2.1.3.1. $D(c, m) < D(c', m)$: c is ranked lower than c', i.e., $c <_{RGB} c'$.

2.1.3.2. $D(c, m) > D(c', m)$: c is ranked higher than c', i.e., $c >_{RGB} c'$.

2.1.3.3. $D(c, m) = D(c', m)$, see subcase 2.3.

2.2. c and c' are lying below V, i.e., $D_{(c,V)} < 0$ and $D_{(c',V)} < 0$; c and c' lie both close to black

Here we can make an analogous reasoning as in case **2.1.** Again we first take the distance to V into account:

2.2.1. $D(c, V) > D(c', V)$: we rank c lower than c', i.e., $c <_{RGB} c'$.

2.2.2. $D(c, V) < D(c', V)$: we rank c higher than c', i.e., $c >_{RGB} c'$.

2.2.3. $D(c, V) = D(c', V) \rightarrow$ the distance to m is taken into account:

2.2.3.1. $D(c, m) > D(c', m)$: c is ranked lower than c', i.e., $c <_{RGB} c'$.

2.2.3.2. $D(c, m) < D(c', m)$: c is ranked higher than c', i.e., $c >_{RGB} c'$.

2.2.3.3. $D(c, m) = D(c', m)$, see subcase 2.3.

2.3. In this part we can take the two subcases **2.1.3.3.** and **2.2.3.3.** together.

If for c and c' hold that $D_{(c,V)} < 0$ and $D_{(c',V)} < 0$ and $D(c,V) = D(c',V)$ and $D(c,m) = D(c',m)$ or $D_{(c,V)} > 0$ and $D_{(c',V)} > 0$ and $D(c,V) = D(c',V)$ and $D(c,m) = D(c',m)$, then both colours c and c' lie not only on the same plane $W_{c,c'}$ parallel to V, but also on the same sphere S at distance $D(c,m) = D(c',m)$ from the centre m. If $D_{(c,V)} = D_{(c',V)} < 0$, the sphere S lies below the plane V; if $D_{(c,V)} = D_{(c',V)} > 0$, the sphere S lies above the plane V. Consequently, c and c' lie on a circle $C_{c,c'}$ in $W_{c,c'}$ parallel to V with centre on the line l. All these colour hues are considered to be equally important, so that we really have to choose one out of these two colours to be the smallest (or largest) colour. And therefore we will order c and c' by defining an angle θ' in $C_{c,c'}$.

If we cut the plane $W_{c,c'}$ with the line l, we get the centre of our circle $C_{c,c'}$. The plane $W_{c,c'}$ is parallel to V, and thus has the same normal vector as V, so we get

$$W_{c,c'} : r + g + b - d_W = 0$$

with $d_W = D_W.\sqrt{3}$, where D_W is the distance of W from the origin $(0,0,0)$.

$$D_W = D_V + D_{(W_{c,c'},V)},$$

where D_V is the distance of $(0,0,0)$ to the plane V, $D_V = \sqrt{\frac{3}{4}} = \frac{\sqrt{3}}{2}$; and $D_{(W_{c,c'},V)}$ is the "distance" between the two planes V and $W_{c,c'}$, $D_{(W_{c,c'},V)} = D_{(c,V)} = D_{(c',V)}$. The centre $a = (r_a, g_a, b_a)$, where $r_a = g_a = b_a$, of the circle $C_{c,c'}$ has to satisfy

$$\begin{cases} r + g + b - d_W &= 0 \\ r &= g \\ g &= b \end{cases},$$

or thus $a = (d_W/3, d_W/3, d_W/3)$. For the radius r of $C_{c,c'}$ we get

$$r = d(c,a) = d(c',a) = \sqrt{(r_c - d_W/3)^2 + (g_c - d_W/3)^2 + (b_c - d_W/3)^2}.$$

Now we will define an angle θ'_c from the centre a of $C_{c,c'}$ for every colour c lying on $C_{c,c'}$:

All lines through a in the plane $W_{c,c'}$ are perpendicular to l. We want to choose a fixed 'direction' that is the same for every plane $W_{c,c'}$, and thus independent of $W_{c,c'}$, through which we define the angle θ'_c. And therefore we cut the plane $W_{c,c'}$ with the upper plane of the cube $b = 1$ so that we get a line $WW_{c,c'}$:

$$\begin{cases} r + g + b - d_W &= 0 \\ b &= 1 \end{cases}$$

or

$$WW_{c,c'} : r + g + (1 - d_W) = 0.$$

Next, we consider the line $X_{c,c'}$ through the centre a perpendicular to the line $WW_{c,c'}$. We determine the equation of $X_{c,c'}$ as the intersection of two planes, namely the plane $W_{c,c'}$ and the plane through the line l perpendicular to the line $WW_{c,c'}$, which has equation $r = g$. So that we obtain for $X_{c,c'}$

$$\left\{ \begin{array}{rcl} r & = & g \\ r+g+b-d_W & = & 0 \end{array} \right. .$$

We know that $X_{c,c'}$ goes through the point a and when we cut $X_{c,c'}$ with the line $WW_{c,c'}$, we get another point γ of $X_{c,c'}$,

$$\left\{ \begin{array}{rcl} r & = & g \\ r+g+b-d_W & = & 0 \\ r+g+(1-d_W) & = & 0 \end{array} \right. ,$$

or thus γ has coordinates $(\frac{d_W-1}{2}, \frac{d_W-1}{2}, 1)$. The line $X_{c,c'}$ gives us the fixed chosen direction, which can be determined for every plane $W_{c,c'}$ in the same way, to define θ'_c. Accordingly, we first define for a colour c lying on the circle $C_{c,c'}$ an angle θ_c determined by

$$\theta_c = \arcsin \frac{d_c}{d(a,c)} \quad (\in [0, \pi/2])$$

with $d(a,c) = \sqrt{(r_c - \frac{d_W}{3})^2 + (g_c - \frac{d_W}{3})^2 + (b_c - \frac{d_W}{3})^2}$, the distance between a and c, and d_c the perpendicular distance between c and the line $X_{c,c'}$ through the two points a and γ. Let $ac(r_{ac}, g_{ac}, b_{ac}) = a(r_a, g_a, b_a) - c(r_c, g_c, b_c)$ and $\gamma a(r_{\gamma a}, g_{\gamma a}, b_{\gamma a}) = \gamma(r_\gamma, g_\gamma, b_\gamma) - a(r_a, g_a, b_a)$. The distance d_c is then given by

$$d_c^2 = \frac{|ac|^2 \, |\gamma a|^2 - ((ac) \cdot (\gamma a))^2}{|\gamma a|^2}$$

or

$$d_c^2 = \frac{(r_{ac}^2 + g_{ac}^2 + b_{ac}^2) \cdot (r_{\gamma a}^2 + g_{\gamma a}^2 + b_{\gamma a}^2) - (r_{ac} \cdot r_{\gamma a} + g_{ac} \cdot g_{\gamma a} + b_{ac} \cdot b_{\gamma a})^2}{(r_{\gamma a}^2 + g_{\gamma a}^2 + b_{\gamma a}^2)}.$$

Now we choose a 'direction' for θ_c and so define the angle θ'_c as

if $b_c > b_a$ (c lies above a)

> if $r_c > g_c$
>
> $\theta'_c = \theta_c$
>
> else if $g_c > r_c$
>
> $\theta'_c = 2\pi - \theta_c$
>
> else if $r_c = g_c$

$\theta'_c = \theta_c = 0$

else if $b_c < b_a$ (c lies below a)

if $r_c > g_c$

$\theta'_c = \pi - \theta_c$

else if $g_c > r_c$

$\theta'_c = \pi + \theta_c$

else if $r_c = g_c$

$\theta'_c = \pi$

else if $b_c = b_a$

if $r_c > g_c$

$\theta'_c = \theta_c = \pi/2$

else if $g_c > r_c$

$\theta'_c = 3\pi/2$

else if $r_c = g_c$

impossible.

In this last step, where for the two colours c and c' hold that $D_{(c,V)} = D_{(c',V)} \neq 0$ and $D(c,m) = D(c',m)$, we order their corresponding angles θ'_c and $\theta'_{c'}$ as follows:

if: (($\theta'_c \in [0, \pi[$ and $\theta'_{c'} \in [0, \pi[$) or ($\theta'_c \in [0, \pi[$ and $\theta'_{c'} \in [\pi, 2\pi[$)) and $\theta'_c < \theta'_{c'}$

then: $\theta'_c <_\theta \theta'_{c'}$

if: $\theta'_c \in [\pi, 2\pi[$ and $\theta'_{c'} \in [\pi, 2\pi[$ and $2\pi - \theta'_c < 2\pi - \theta'_{c'}$ (or thus $\theta'_c > \theta'_{c'}$)

then: $\theta'_c <_\theta \theta'_{c'}$.

2.4. c and c' are lying in V, i.e., $D_{(c,V)} = D_{(c',V)} = 0$

Because we want our ordering to be compatible with the complement *co*, we order the colours c and c' here as follows:

2.4.1. $D(c,m)\cos(\theta'_c) < D(c',m)\cos(\theta'_{c'})$: we rank c lower than c', i.e., $c <_{RGB} c'$.

2.4.2. $D(c,m)\cos(\theta'_c) > D(c',m)\cos(\theta'_{c'})$: we rank c higher than c', i.e., $c >_{RGB} c'$.

2.4.3. $D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'})$ and $D(c,m)\sin(\theta'_c) < D(c',m)\sin(\theta'_{c'})$: we rank c lower than c', i.e., $c <_{RGB} c'$.

2.4.4. $D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'})$ and $D(c,m)\sin(\theta'_c) > D(c',m)\sin(\theta'_{c'})$: we rank c higher than c', i.e., $c >_{RGB} c'$.

Definition of the new RGB colour vector ordering $\leq_{RGB}$

Consider two colours $c(r, g, b)$ and $c'(r', g', b')$ in RGB, then it holds that

$$
\begin{aligned}
c <_{RGB} c' \;\Leftrightarrow\; & \big(D_{(c,V)} < 0 \text{ and } D_{(c',V)} > 0\big) \\
& \text{or } \big(D_{(c,V)} = 0 \text{ and } D_{(c',V)} > 0\big) \\
& \text{or } \big(D_{(c,V)} < 0 \text{ and } D_{(c',V)} = 0\big) \\
& \text{or } \big(D_{(c,V)} < 0 \text{ and } D_{(c',V)} < 0 \text{ and } D(c,V) > D(c',V)\big) \\
& \text{or } \big(D_{(c,V)} > 0 \text{ and } D_{(c',V)} > 0 \text{ and } D(c,V) < D(c',V)\big) \\
& \text{or } \big(D_{(c,V)} < 0 \text{ and } D_{(c',V)} < 0 \text{ and } D(c,V) = D(c',V) \\
& \qquad \text{and } D(c,m) > D(c',m)\big) \\
& \text{or } \big(D_{(c,V)} > 0 \text{ and } D_{(c',V)} > 0 \text{ and } D(c,V) = D(c',V) \\
& \qquad \text{and } D(c,m) < D(c',m)\big) \\
& \text{or } \big(D_{(c,V)} = D_{(c',V)} \neq 0 \text{ and } D(c,m) = D(c',m) \text{ and } \theta'_c <_\theta \theta'_{c'}\big) \\
& \text{or } (D_{(c,V)} = D_{(c',V)} = 0 \text{ and } D(c,m)\cos(\theta'_c) < D(c',m)\cos(\theta'_{c'})) \\
& \text{or } (D_{(c,V)} = D_{(c',V)} = 0 \text{ and } D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'}) \\
& \qquad \text{and } D(c,m)\sin(\theta'_c) < D(c',m)\sin(\theta'_{c'}))
\end{aligned}
$$

$$c >_{RGB} c' \;\Leftrightarrow\; c' <_{RGB} c$$

$$c =_{RGB} c' \;\Leftrightarrow\; D_{(c,V)} = D_{(c',V)} \text{ and } D(c,m) = D(c',m) \text{ and } \theta'_c = \theta'_{c'}$$

$$c \leq_{RGB} c' \;\Leftrightarrow\; c <_{RGB} c' \text{ or } c =_{RGB} c'.$$

Properties of $\leq_{RGB}$

We examine some properties of our new ordering $\leq_{RGB}$.

1. Reflexive: $(\forall a \in RGB)(a \leq_{RGB} a)$. OK.

2. Antisymmetric: $(\forall a, b \in RGB)(a \leq_{RGB} b \text{ and } b \leq_{RGB} a \overset{?}{\Rightarrow} a =_{RGB} b)$

Proof

Suppose that $a \neq_{RGB} b$.

1.1 $(D_{(a,V)} < 0$ and $D_{(b,V)} > 0)$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow: $D_{(a,V)} > 0$, a contradiction.

1.2 $(D_{(a,V)} = 0$ and $D_{(b,V)} > 0)$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow: $D_{(a,V)} > 0$, a contradiction.

1.3 $(D_{(a,V)} < 0$ and $D_{(b,V)} = 0)$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow: $D_{(a,V)} \geq 0$, a contradiction.

2.1.1 $(D_{(a,V)} > 0$ and $D_{(b,V)} > 0$ and $D(a,V) < D(b,V))$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:
$D_{(a,V)} > 0$ and $D(b,V) \leq D(a,V)$, a contradiction.

2.2.1 $(D_{(a,V)} < 0$ and $D_{(b,V)} < 0$ and $D(a,V) > D(b,V))$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:

(a) $D_{(b,V)} < 0$ and $D_{(a,V)} \geq 0$, a contradiction.

(b) $D_{(b,V)} < 0$ and $D_{(a,V)} < 0$ and $D(b,V) \geq D(a,V)$, a contradiction.

2.1.3 $(D_{(a,V)} > 0$ and $D_{(b,V)} > 0$ and $D(a,V) = D(b,V)$ and $D(a,m) < D(b,m))$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:
$D(b,m) \leq D(a,m)$, a contradiction.

2.2.3 $(D_{(a,V)} < 0$ and $D_{(b,V)} < 0$ and $D(a,V) = D(b,V)$ and $D(a,m) > D(b,m))$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:

(a) $D_{(b,V)} < 0$ and $D_{(a,V)} \geq 0$, a contradiction.

(b) $D_{(b,V)} < 0$ and $D_{(a,V)} < 0$ and $D(a,V) = D(b,V)$ and $D(b,m) \geq D(a,m)$, a contradiction.

2.3 $(D_{(a,V)} = D_{(b,V)} \neq 0$ and $D(a,m) = D(b,m)$ and $\theta'_a <_\theta \theta'_b)$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow: $\theta'_b <_\theta \theta'_a$, a contradiction.

2.4.1 $(D_{(a,V)} = D_{(b,V)} = 0$ and $D(a,m)\cos(\theta'_a) < D(b,m)\cos(\theta'_b))$ and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:
$D(b,m)\cos(\theta'_b) \leq D(a,m)\cos(\theta'_a)$, a contradiction.

2.4.3 ($D_{(a,V)} = D_{(b,V)} = 0$ and $D(a,m)\cos(\theta'_a) = D(b,m)\cos(\theta'_b)$ and $D(a,m)\sin(\theta'_a) < D(b,m)\sin(\theta'_b)$) and $b <_{RGB} a$

From the definition of $<_{RGB}$ would follow:
$D(b,m)\sin(\theta'_b) < D(a,m)\sin(\theta'_a)$, a contradiction.

$\Rightarrow$ $D_{(a,V)} = D_{(b,V)}$ and $D(a,m) = D(b,m)$ and $\theta'_a = \theta'_b$

$\Rightarrow$ $a =_{RGB} b$.

■

3. Transitive: $(\forall a,b,c \in RGB)(a \leq_{RGB} b \text{ and } b \leq_{RGB} c \stackrel{?}{\Rightarrow} a \leq_{RGB} c)$

Proof

Let $a =_{RGB} b$ and $b \leq_{RGB} c$, then it holds that $a \leq_{RGB} c$. Let $a \leq_{RGB} b$ and $b =_{RGB} c$, then it holds that $a \leq_{RGB} c$. So suppose that $a \neq_{RGB} b$ and $b \neq_{RGB} c$, thus $a <_{RGB} b$ and $b <_{RGB} c$.

1.1 ($D_{(a,V)} < 0$ and $D_{(b,V)} > 0$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow: $D_{(b,V)} > 0$ and $D_{(c,V)} > 0$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} > 0$

$\Rightarrow$ $a <_{RGB} c$.

1.2 ($D_{(a,V)} = 0$ and $D_{(b,V)} > 0$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow: $D_{(b,V)} > 0$ and $D_{(c,V)} > 0$
$\Rightarrow$ $D_{(a,V)} = 0$ and $D_{(c,V)} > 0$

$\Rightarrow$ $a <_{RGB} c$.

1.3 ($D_{(a,V)} < 0$ and $D_{(b,V)} = 0$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow: $D_{(b,V)} = 0$ and $D_{(c,V)} \geq 0$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} \geq 0$

$\Rightarrow$ $a <_{RGB} c$.

2.1.1 ($D_{(a,V)} > 0$ and $D_{(b,V)} > 0$ and $D(a,V) < D(b,V)$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow:
$D_{(b,V)} > 0$ and $D_{(c,V)} > 0$ and $D(b,V) \leq D(c,V)$
$\Rightarrow$ $D_{(a,V)} > 0$ and $D_{(c,V)} > 0$ and $D(a,V) < D(c,V)$

$\Rightarrow$ $a <_{RGB} c$.

2.2.1 ($D_{(a,V)} < 0$ and $D_{(b,V)} < 0$ and $D(a,V) > D(b,V)$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow:

(a) $D_{(b,V)} < 0$ and $D_{(c,V)} \geq 0$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} \geq 0$

(b) $D_{(b,V)} < 0$ and $D_{(c,V)} < 0$ and $D(b,V) \geq D(c,V)$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} < 0$ and $D(a,V) > D(c,V)$

$\Rightarrow$ $a <_{RGB} c$.

2.1.3 ($D_{(a,V)} > 0$ and $D_{(b,V)} > 0$ and $D(a,V) = D(b,V)$ and $D(a,m) < D(b,m)$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow that:

(a) $D_{(b,V)} > 0$ and $D_{(c,V)} > 0$ and $D(b,V) < D(c,V)$
$\Rightarrow$ $D_{(a,V)} > 0$ and $D_{(c,V)} > 0$ and $D(a,V) < D(c,V)$

(b) $D_{(b,V)} > 0$ and $D_{(c,V)} > 0$ and $D(b,V) = D(c,V)$ and $D(b,m) \leq D(c,m)$
$\Rightarrow$ $D_{(a,V)} > 0$ and $D_{(c,V)} > 0$ and $D(a,V) = D(c,V)$ and $D(a,m) < D(c,m)$

$\Rightarrow$ $a <_{RGB} c$.

2.2.3 ($D_{(a,V)} < 0$ and $D_{(b,V)} < 0$ and $D(a,V) = D(b,V)$ and $D(a,m) > D(b,m)$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow that:

(a) $D_{(b,V)} < 0$ and $D_{(c,V)} \geq 0$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} \geq 0$

(b) $D_{(b,V)} < 0$ and $D_{(c,V)} < 0$ and $D(b,V) > D(c,V)$
$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} < 0$ and $D(a,V) > D(c,V)$

(c) $D_{(b,V)} < 0$ and $D_{(c,V)} < 0$ and $D(b,V) = D(c,V)$ and $D(b,m) \geq D(c,m)$

$\Rightarrow$ $D_{(a,V)} < 0$ and $D_{(c,V)} < 0$ and $D(a,V) = D(c,V)$ and $D(a,m) > D(c,m)$

$\Rightarrow$ $a <_{RGB} c$.

2.3 ($D_{(a,V)} = D_{(b,V)} \neq 0$ and $D(a,m) = D(b,m)$ and $\theta'_a <_\theta \theta'_b$) and $b <_{RGB} c$.

From the definition of $<_{RGB}$ would follow that:

(a) $D_{(a,V)} = D_{(b,V)} < 0$

1. $D_{(c,V)} \geq 0$
2. $D_{(c,V)} < 0$ and $D(b,V) > D(c,V)$
3. $D_{(c,V)} < 0$ and $D(b,V) = D(c,V)$ and $D(b,m) > D(c,m)$
4. $D_{(b,V)} = D_{(c,V)}$ and $D(b,m) = D(c,m)$ and $\theta'_b <_\theta \theta'_c$

(b) $D_{(a,V)} = D_{(b,V)} > 0$

1. $D_{(c,V)} > 0$ and $D(b,V) < D(c,V)$
2. $D_{(c,V)} > 0$ and $D(b,V) = D(c,V)$ and $D(b,m) < D(c,m)$
3. $D_{(b,V)} = D_{(c,V)}$ and $D(b,m) = D(c,m)$ and $\theta'_b <_\theta \theta'_c$

$\Rightarrow$ $a <_{RGB} c$

2.4.1 ($D_{(a,V)} = D_{(b,V)} = 0$ and $D(a,m)\cos(\theta'_a) < D(b,m)\cos(\theta'_b)$) and $b <_{RGB} c$

From the definition of $<_{RGB}$ would follow:

(a) $D_{(c,V)} > 0$

(b) $D_{(b,V)} = D_{(c,V)}$ and $D(b,m)\cos(\theta'_b) \leq D(c,m)\cos(\theta'_c)$

$\Rightarrow$ $a <_{RGB} c$

2.4.3 ($D_{(a,V)} = D_{(b,V)} = 0$ and $D(a,m)\cos(\theta'_a) = D(b,m)\cos(\theta'_b)$ and $D(a,m)\sin(\theta'_a) < D(b,m)\sin(\theta'_b)$) and $b <_{RGB} c$

From the definition of $<_{RGB}$ would follow:

(a) $D_{(c,V)} > 0$

(b) $D_{(b,V)} = D_{(c,V)}$ and $D(b,m)\cos(\theta'_b) < D(c,m)\cos(\theta'_c)$

(c) $D_{(b,V)} = D_{(c,V)}$ and $D(b,m)\cos(\theta'_b) = D(c,m)\cos(\theta'_c)$ and $D(b,m)\sin(\theta'_b) < D(c,m)\sin(\theta'_c)$

$\Rightarrow$ $a \leq_{RGB} c$.

■

We now prove that our new ordering $\leq_{RGB}$ is compatible with the complement co, i.e.,

$$\text{for all colours } c, c' \text{ in } RGB : c \leq_{RGB} c' \stackrel{?}{\Leftrightarrow} co(c) \geq_{RGB} co(c').$$

Proof

1.1 $(D_{(c,V)} < 0 \text{ and } D_{(c',V)} > 0)$

$$\begin{array}{rcl} D_{(c,V)} < 0 & \Leftrightarrow & r_c + g_c + b_c < 3/2 \\ & \Leftrightarrow & 3 - (r_c + g_c + b_c) > 3 - 3/2 \\ & \Leftrightarrow & (1 - r_c) + (1 - g_c) + (1 - b_c) > 3/2 \\ & \Leftrightarrow & D_{(co(c),V)} > 0 \end{array}$$

So we get

$$(D_{(c,V)} < 0 \text{ and } D_{(c',V)} > 0) \Leftrightarrow (D_{(co(c),V)} > 0 \text{ and } D_{(co(c'),V)} < 0).$$

1.2

$$(D_{(c,V)} = 0 \text{ and } D_{(c',V)} > 0) \quad \Leftrightarrow \quad (D_{(co(c),V)} = 0 \text{ and } D_{(co(c'),V)} < 0).$$

1.3

$$(D_{(c,V)} < 0 \text{ and } D_{(c',V)} = 0) \quad \Leftrightarrow \quad (D_{(co(c),V)} > 0 \text{ and } D_{(co(c'),V)} = 0).$$

2.1.1 $(D_{(c,V)} > 0 \text{ and } D_{(c',V)} > 0)$ and $D(c,V) < D(c',V)$, where

$$\begin{array}{rcl} D(c,V) & = & \frac{|r_c+g_c+b_c-3/2|}{\sqrt{3}} \\ & = & \frac{|-(r_c+g_c+b_c)+3/2|}{\sqrt{3}} \\ & = & \frac{|3-(r_c+g_c+b_c)-3/2|}{\sqrt{3}} \\ & = & \frac{|(1-r_c)+(1-g_c)+(1-b_c)-3/2|}{\sqrt{3}} \\ & = & D(co(c),V). \end{array}$$

So we get
$(D_{(c,V)} > 0 \text{ and } D_{(c',V)} > 0)$ and $D(c,V) < D(c',V) \Leftrightarrow$
$(D_{(co(c),V)} < 0 \text{ and } D_{(co(c'),V)} < 0)$ and $D(co(c),V) < D(co(c'),V)$.

2.2.1 $(D_{(c,V)} < 0 \text{ and } D_{(c',V)} < 0)$ and $D(c,V) > D(c',V) \Leftrightarrow$
$(D_{(co(c),V)} > 0 \text{ and } D_{(co(c'),V)} > 0)$ and $D(co(c),V) > D(co(c'),V)$.

2.1.3 ($D_{(c,V)} > 0$ and $D_{(c',V)} > 0$ and $D(c,V) = D(c',V)$) and $D(c,m) < D(c',m)$, where

$$\begin{aligned}
D(c,m) < D(c',m) \quad &\Leftrightarrow \quad (r_c - 1/2)^2 + (g_c - 1/2)^2 + (b_c - 1/2)^2 < \\
&\quad\; (r_{c'} - 1/2)^2 + (g_{c'} - 1/2)^2 + (b_{c'} - 1/2)^2 \\
&\Leftrightarrow \quad r_c^2 - r_c + g_c^2 - g_c + b_c^2 - b_c < \\
&\quad\; r_{c'}^2 - r_{c'} + g_{c'}^2 - g_{c'} + b_{c'}^2 - b_{c'} \\
&\Leftrightarrow \quad 1 - 2r_c + r_c^2 - 1 + r_c + 1 - 2g_c + g_c^2 - 1 + g_c \\
&\quad\; +1 - 2b_c + b_c^2 - 1 + b_c < \\
&\quad\; 1 - 2r_{c'} + r_{c'}^2 - 1 + r_{c'} + 1 - 2g_{c'} + g_{c'}^2 - 1 \\
&\quad\; +g_{c'} + 1 - 2b_{c'} + b_{c'}^2 - 1 + b_{c'} \\
&\Leftrightarrow \quad (1 - r_c)^2 + 1/4 - (1 - r_c) + (1 - g_c)^2 + 1/4 \\
&\quad\; -(1 - g_c) + (1 - b_c)^2 + 1/4 - (1 - b_c) < \\
&\quad\; (1 - r_{c'})^2 + 1/4 - (1 - r_{c'}) + (1 - g_{c'})^2 + 1/4 \\
&\quad\; -(1 - g_{c'}) + (1 - b_{c'})^2 + 1/4 - (1 - b_{c'}) \\
&\Leftrightarrow \quad ((1 - r_c) - 1/2)^2 + ((1 - g_c) - 1/2)^2 + \\
&\quad\; ((1 - b_c) - 1/2)^2) < ((1 - r_{c'}) - 1/2)^2 + \\
&\quad\; ((1 - g_{c'}) - 1/2)^2 + ((1 - b_{c'}) - 1/2)^2) \\
&\Leftrightarrow \quad D(co(c), m) < D(co(c'), m)
\end{aligned}$$

So we get ($D_{(c,V)} > 0$ and $D_{(c',V)} > 0$) and $D(c,V) = D(c',V)$ and $D(c,m) < D(c',m) \Leftrightarrow (D_{(co(c),V)} < 0$ and $D_{(co(c'),V)} < 0)$ and $D(co(c),V) = D(co(c'),V)$ and $D(co(c),m) < D(co(c'),m)$.

2.2.3 ($D_{(c,V)} < 0$ and $D_{(c',V)} < 0$) and $D(c,V) = D(c',V)$ and $D(c,m) > D(c',m) \Leftrightarrow (D_{(co(c),V)} > 0$ and $D_{(co(c'),V)} > 0)$ and $D(co(c),V) = D(co(c'),V)$ and $D(co(c),m) > D(co(c'),m)$.

2.3 $D_{(c,V)} = D_{(c',V)} \neq 0$ and $D(c,m) = D(c',m)$

A. $\theta'_c \in [0, \pi[$

i. $\theta'_{c'} \in [0, \pi[$

From $c \leq_{RGB} c'$ it follows that $\theta'_c \leq \theta'_{c'}$. By definition of co we get: $\theta'_{co(c)} = \theta'_c + \pi$ and $\theta'_{co(c')} = \theta'_{c'} + \pi$ and hence from $\theta'_c \leq \theta'_{c'}$ we get $\theta'_c + \pi \leq \theta'_{c'} + \pi$, i.e., $\theta'_{co(c)} \leq \theta'_{co(c')}$.

ii. $\theta'_{c'} \in [\pi, 2\pi[$

From $c \leq_{RGB} c'$ it follows that $\theta'_c \leq \theta'_{c'}$. By definition of co we get: $\theta'_{co(c)} = \theta'_c + \pi$ and $\theta'_{co(c')} = \theta'_{c'} - \pi$ and hence from $\theta'_c + \pi \in [\pi, 2\pi[$ and $\theta'_{c'} - \pi \in [0, \pi[$ we get $\theta'_{co(c')} \leq \theta'_{co(c)}$.

B. $\theta'_c \in [\pi, 2\pi[$

i. $\theta'_{c'} \in [0, \pi[$, impossible

ii. $\theta'_{c'} \in [\pi, 2\pi[$
From $c \leq_{RGB} c'$ it follows that $\theta'_c \geq \theta'_{c'}$. By definition of co we get: $\theta'_{co(c)} = \theta'_c - \pi$ and $\theta'_{co(c')} = \theta'_{c'} - \pi$ and hence from $\theta'_c \geq \theta'_{c'}$ we get $\theta'_c - \pi \geq \theta'_{c'} - \pi$, i.e., $\theta'_{co(c)} \geq \theta'_{co(c')}$.

2.4.1 $D_{(c,V)} = D_{(c',V)} = 0$ and $D(c,m)\cos(\theta'_c) < D(c',m)\cos(\theta'_{c'})$, where

$$
\begin{aligned}
& D(c,m)\cos(\theta'_c) < D(c',m)\cos(\theta'_{c'}) \\
\Leftrightarrow\ & -D(c,m)\cos(\theta'_c) > -D(c',m)\cos(\theta'_{c'}) \\
\Leftrightarrow\ & D(c,m)\cos((\theta'_c + \pi) \bmod 2\pi) > D(c',m)\cos((\theta'_{c'} + \pi) \bmod 2\pi) \\
\Leftrightarrow\ & D(c,m)\cos(\theta'_{co(c)}) > D(c',m)\cos(\theta'_{co(c')}),
\end{aligned}
$$

where we have used the property that

$$\cos(x + \pi) = \cos(x - \pi) = -\cos(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.
So we get $D_{(c,V)} = D_{(c',V)} = 0$ and $D(c,m)\cos(\theta'_c) < D(c',m)\cos(\theta'_{c'}) \Leftrightarrow$ $D_{(co(c),V)} = D_{(co(c'),V)} = 0$ and $D(c,m)\cos(\theta'_{co(c)}) > D(c',m)\cos(\theta'_{co(c')})$.

2.4.3 $D_{(c,V)} = D_{(c',V)} = 0$ and $D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'})$ and $D(c,m)\sin(\theta'_c) < D(c',m)\sin(\theta'_{c'})$, where

$$
\begin{aligned}
& D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'}) \\
\Leftrightarrow\ & -D(c,m)\cos(\theta'_c) = -D(c',m)\cos(\theta'_{c'}) \\
\Leftrightarrow\ & D(c,m)\cos((\theta'_c + \pi) \bmod 2\pi) = D(c',m)\cos((\theta'_{c'} + \pi) \bmod 2\pi) \\
\Leftrightarrow\ & D(c,m)\cos(\theta'_{co(c)}) = D(c',m)\cos(\theta'_{co(c')})
\end{aligned}
$$

and

$$
\begin{aligned}
& D(c,m)\sin(\theta'_c) < D(c',m)\sin(\theta'_{c'}) \\
\Leftrightarrow\ & -D(c,m)\sin(\theta'_c) > -D(c',m)\sin(\theta'_{c'}) \\
\Leftrightarrow\ & D(c,m)\sin((\theta'_c + \pi) \bmod 2\pi) > D(c',m)\sin((\theta'_{c'} + \pi) \bmod 2\pi) \\
\Leftrightarrow\ & D(c,m)\sin(\theta'_{co(c)}) > D(c',m)\sin(\theta'_{co(c')}),
\end{aligned}
$$

where we have used the property that

$$\sin(x + \pi) = \sin(x - \pi) = -\sin(x)$$

and

$$(x + \pi) \bmod 2\pi = x - \pi \text{ or } x + \pi$$

for all $x \in [0, 2\pi]$.
So we get $D_{(c,V)} = D_{(c',V)} = 0$ and $D(c,m)\cos(\theta'_c) = D(c',m)\cos(\theta'_{c'})$ and $D(c,m)\sin(\theta'_c) < D(c',m)\sin(\theta'_{c'}) \Leftrightarrow D_{(co(c),V)} = D_{(co(c'),V)} = 0$ and $D(c,m)\cos(\theta'_{co(c)}) = D(c',m)\cos(\theta'_{co(c')})$ and $D(c,m)\sin(\theta'_{co(c)}) > D(c',m)\sin(\theta'_{co(c')})$.

$$\Rightarrow c \leq_{RGB} c' \overset{!}{\Leftrightarrow} co(c) \geq_{RGB} co(c').$$

■

4.3.8 Associated Minimum and Maximum Operators

The minimum (maximum) of a set S of n colours $c_1(r_1, g_1, b_1), \ldots, c_n(r_n, g_n, b_n)$ in RGB is the colour $c_\alpha \in S$ wherefore $c_\alpha \leq_{RGB} c_i$ ($c_\alpha \geq_{RGB} c_i$), for all $i = 1 \ldots n$.

$(RGB, \leq_{RGB})$ is a poset, and what is more, by definition of the order relation $\leq_{RGB}$, it holds that

$$(\forall c, c' \in RGB)(c \leq_{RGB} c' \text{ or } c' \leq_{RGB} c),$$

so that $(RGB, \leq_{RGB})$ is a totally ordered set ($\max_{RGB}(c, c')$ and $\min_{RGB}(c, c')$ exist for all $c, c' \in RGB$), and thus a lattice. The greatest element in $(RGB, \leq_{RGB})$ is $\mathbf{1} = (1, 1, 1)$ and the smallest element is $\mathbf{0} = (0, 0, 0)$, so we get a bounded complete lattice. We will sometimes drop the index RGB.

Since colour images in the RGB colour model can be modelled as $\mathbb{R}^2 - (RGB, \leq_{RGB})$ mappings and because $(RGB, \leq_{RGB})$ is a complete lattice, we can identify colour images in RGB with $\mathcal{L}$-fuzzy sets on $\mathbb{R}^2$, with $(\mathcal{L}, \leq_{\mathcal{L}}) = (RGB, \leq_{RGB})$, and thus define for a family $(A_i)_{i=1}^n$ of colour images in RGB

$$\begin{aligned} \bigcap_{i=1}^{n}{}_{RGB} A_i(x) &= \min_{i=1\ldots n}{}_{RGB} A_i(x), \ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n}{}_{RGB} A_i(x) &= \max_{i=1\ldots n}{}_{RGB} A_i(x), \ \forall x \in \mathbb{R}^2, \end{aligned}$$

so that $(\mathcal{F}_{RGB}(X), \cap_{RGB}, \cup_{RGB})$ is a lattice with an ordering defined for all $A, B \in \mathcal{F}_{RGB}(X)$ as

$$A \subseteq_{RGB} B \Leftrightarrow (\forall x \in X)(A(x) \leq_{RGB} B(x)).$$

Let $\mathcal{N}$ be a negator on RGB, $\mathcal{T}$ a t-norm and $\mathcal{S}$ a t-conorm on RGB. For a colour image A in RGB and a family $(A_i)_{i=1}^n$ of colour images in RGB we define

$$\begin{aligned} co_{\mathcal{N}}A(x) &= \mathcal{N}(A(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcap_{i=1}^{n}{}_{\mathcal{T}}\, A_i(x) &= \mathcal{T}(A_1(x), A_2(x), \ldots, 1A_n(x)),\ \forall x \in \mathbb{R}^2, \\ \bigcup_{i=1}^{n}{}_{\mathcal{S}}\, A_i(x) &= \mathcal{S}(A_1(x), A_2(x), \ldots, A_n(x)),\ \forall x \in \mathbb{R}^2. \end{aligned}$$

Extension of greyscale morphology to colour morphology in RGB

We extend the basic morphological operators dilation and erosion for greyscale images based on the threshold, umbra and fuzzy approach to colour images modelled in RGB. For proofs we refer to the equivalent properties in the HSV colour model.

• Threshold approach

Let A be a colour image, represented as a $\mathbb{R}^2 - (RGB, \leq_{RGB})$ mapping, and B a binary structuring element ($\subseteq \mathbb{R}^2$).

Definition 4.51. *Let A be a colour image and B a binary structuring element. The **threshold 'colour' dilation** $\vec{D}_t(A,B)$ and the **threshold 'colour' erosion** $\vec{E}_t(A,B)$ are the colour images given by*

$$\vec{D}_t(A,B)(y) \stackrel{def}{=}_{RGB} \max_{x \in T_y(B)} A(x) \qquad \textit{for } y \in \mathbb{R}^2,$$

$$\vec{E}_t(A,B)(y) \stackrel{def}{=}_{RGB} \min_{x \in T_y(B)} A(x) \qquad \textit{for } y \in \mathbb{R}^2.$$

Property 4.52.

$$\begin{aligned} \vec{D}_t(\overline{\mathbf{0}}, B) =_{RGB} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(\overline{\mathbf{1}}, B) =_{RGB} \overline{\mathbf{1}} \\ \vec{D}_t(A, \emptyset) =_{RGB} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E}_t(A, \emptyset) =_{RGB} \overline{\mathbf{1}}. \end{aligned}$$

Property 4.53 (Duality dilation-erosion).

$$\begin{aligned} \vec{D}_t(A,B) &=_{RGB} co(\vec{E}_t(co(A), B)) \\ \vec{E}_t(A,B) &=_{RGB} co(\vec{D}_t(co(A), B)). \end{aligned}$$

Property 4.54 (Monotonicity). *If A and B are two colour images, and C and C' are two binary structuring elements, then it holds that*

$$\begin{aligned} A \subseteq_{RGB} B \quad &\Rightarrow \quad \vec{D}_t(A,C) \subseteq_{RGB} \vec{D}_t(B,C) \text{ and} \\ &\quad\ \vec{E}_t(A,C) \subseteq_{RGB} \vec{E}_t(B,C) \\ C \subseteq C' \quad &\Rightarrow \quad \vec{D}_t(A,C) \subseteq_{RGB} \vec{D}_t(A,C') \text{ and} \\ &\quad\ \vec{E}_t(A,C) \supseteq_{RGB} \vec{E}_t(A,C'). \end{aligned}$$

Property 4.55 (Inclusion).

$$\vec{E}_t(A,B) \subseteq_{RGB} \vec{D}_t(A,B).$$

Property 4.56 (Extensity).

$$\mathbf{0} \in B \quad \Rightarrow \quad A \subseteq_{RGB} \vec{D}_t(A,B) \text{ and } \vec{E}_t(A,B) \subseteq_{RGB} A.$$

Property 4.57 (Interaction with intersection and union). *Consider a family $(A_i)_{i=1}^n$ of colour images and a family $(B_i)_{i=1}^n$ of binary structuring elements.*
For the t-'colour' dilation it holds that

$$\begin{aligned} \vec{D}_t(\bigcap_{i=1}^n A_i, B) &\subseteq_{RGB} \bigcap_{i=1}^n \vec{D}_t(A_i,B) \\ \vec{D}_t(A, \bigcap_{i=1}^n B_i) &\subseteq_{RGB} \bigcap_{i=1}^n \vec{D}_t(A,B_i); \\ \vec{D}_t(\bigcup_{i=1}^n A_i, B) &=_{RGB} \bigcup_{i=1}^n \vec{D}_t(A_i,B) \\ \vec{D}_t(A, \bigcup_{i=1}^n B_i) &=_{RGB} \bigcup_{i=1}^n \vec{D}_t(A,B_i). \end{aligned}$$

For the t-'colour' erosion it holds that

$$\begin{aligned} \vec{E}_t(\bigcap_{i=1}^n A_i, B) &=_{RGB} \bigcap_{i=1}^n \vec{E}_t(A_i,B) \\ \vec{E}_t(A, \bigcap_{i=1}^n B_i) &\supseteq_{RGB} \bigcup_{i=1}^n \vec{E}_t(A,B_i); \\ \vec{E}_t(\bigcup_{i=1}^n A_i, B) &\supseteq_{RGB} \bigcup_{i=1}^n \vec{E}_t(A_i,B) \\ \vec{E}_t(A, \bigcup_{i=1}^n B_i) &=_{RGB} \bigcap_{i=1}^n \vec{E}_t(A,B_i). \end{aligned}$$

Property 4.58. *Let A be a colour image and B and C two binary structuring elements, then*

$$\begin{aligned}\vec{D}_t(\vec{D}_t(A,B),C) &=_{RGB} \vec{D}_t(\vec{D}_t(A,C),B)\\ \vec{E}_t(\vec{E}_t(A,B),C) &=_{RGB} \vec{E}_t(\vec{E}_t(A,C),B).\end{aligned}$$

• Fuzzy approach

The support d_A of a colour image A in RGB is defined as the set

$$d_A = \{x \in \mathbb{R}^2 \mid A(x) >_{RGB} \mathbf{0}\}.$$

Definition 4.59. *Let A be a colour image and B a colour structuring element (both seen as $(RGB, \leq_{RGB})$-fuzzy sets), $\mathcal{C}$ a conjunctor on $(RGB, \leq_{RGB})$ and $\mathcal{I}$ an implicator on $(RGB, \leq_{RGB})$. The **fuzzy 'colour' dilation** $\vec{D}_{\mathcal{C}}(A,B)$ and the **fuzzy 'colour' erosion** $\vec{E}_{\mathcal{I}}(A,B)$ are the $(RGB, \leq_{RGB})$-fuzzy sets defined as*

$$\begin{aligned}\vec{D}_{\mathcal{C}}(A,B)(y) &\stackrel{def}{=}_{RGB} \max_{x \in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) \quad \text{for } y \in \mathbb{R}^2,\\ \vec{E}_{\mathcal{I}}(A,B)(y) &\stackrel{def}{=}_{RGB} \min_{x \in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) \quad \text{for } y \in \mathbb{R}^2.\end{aligned}$$

Notice that we can write

$$\begin{aligned}\max_{x \in T_y(d_B)} \mathcal{C}(B(x-y), A(x)) &\stackrel{def}{=}_{RGB} \max_{x \in \mathbb{R}^2} \mathcal{C}(R_B(x,y), A(x)),\\ \min_{x \in T_y(d_B)} \mathcal{I}(B(x-y), A(x)) &\stackrel{def}{=}_{RGB} \min_{x \in \mathbb{R}^2} \mathcal{I}(R_B(x,y), A(x)),\end{aligned}$$

for all y in $\mathbb{R}^2$, whereby

$$R_B : \mathbb{R}^2 \times \mathbb{R}^2 \to (RGB, \leq_{RGB}),$$

$R_B = B \circ V$ with V defined as $V(x,y) = x - y = (x_1, x_2) - (y_1, y_2) = (x_1 - y_1, x_2 - y_2),\ \forall (x,y) \in (\mathbb{R}^2)^2$ so that $R_B(x,y) = B(x-y)$.

Property 4.60. *[12] Let $\mathcal{T}$ be a t-norm and $\mathcal{I}$ be an implicator on $(RGB, \leq_{RGB})$, then it holds:*

$$\vec{D}_{\mathcal{C}}(\overline{\mathbf{0}}, B) =_{RGB} \overline{\mathbf{0}} \quad \text{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{1}}, B) =_{RGB} \overline{\mathbf{1}}.$$

If $B(\mathbf{0}) =_{RGB} \mathbf{1}$, then it holds:

$$\vec{D}_{\mathcal{C}}(\overline{\mathbf{1}}, B) =_{RGB} \overline{\mathbf{1}} \quad \text{and} \quad \vec{E}_{\mathcal{I}}(\overline{\mathbf{0}}, B) =_{RGB} \overline{\mathbf{0}}.$$

Property 4.61 (Duality dilation-erosion). *[12] Let $\mathcal{T}$ be a t-norm on RGB, $\mathcal{N}$ an involutive negator on RGB and $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ the corresponding $\mathcal{S}$-implicator. For every colour image A and colour structuring element B we have*

$$\begin{aligned} co_{\mathcal{N}}\vec{D_{\mathcal{T}}}(A,B) &=_{RGB} \vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(co_{\mathcal{N}}A,B) \\ \vec{D_{\mathcal{T}}}(co_{\mathcal{N}}A,B) &=_{RGB} co_{\mathcal{N}}(\vec{E}_{\mathcal{I}_{\mathcal{T},\mathcal{N}}}(A,B)). \end{aligned}$$

Property 4.62 (Monotonicity). *[Generalisation of [12]] If A and B are two colour images, C and C' two colour structuring elements, $\mathcal{C}_1$ and $\mathcal{C}_2$ two conjunctors and $\mathcal{I}_1$ and $\mathcal{I}_2$ two implicators on RGB, then it holds that*

$$\begin{aligned} A \subseteq_{RGB} B \quad &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{RGB} \vec{D_{\mathcal{C}}}(B,C) \text{ and} \\ &\qquad\quad \vec{E_{\mathcal{I}}}(A,C) \subseteq_{RGB} \vec{E_{\mathcal{I}}}(B,C) \\ C \subseteq_{RGB} C' \quad &\Rightarrow \quad \vec{D_{\mathcal{C}}}(A,C) \subseteq_{RGB} \vec{D_{\mathcal{C}}}(A,C') \text{ and} \\ &\qquad\quad \vec{E_{\mathcal{I}}}(A,C) \supseteq_{RGB} \vec{E_{\mathcal{I}}}(A,C') \\ \mathcal{C}_1 \subseteq_{RGB} \mathcal{C}_2 \quad &\Rightarrow \quad \vec{D_{\mathcal{C}_1}}(A,C) \subseteq_{RGB} \vec{D_{\mathcal{C}_2}}(A,C) \text{ and} \\ \mathcal{I}_1 \subseteq_{RGB} \mathcal{I}_2 \quad &\Rightarrow \quad \vec{E_{\mathcal{I}_1}}(A,C) \subseteq_{RGB} \vec{E_{\mathcal{I}_2}}(A,C) \end{aligned}$$

Property 4.63 (Inclusion). *[Generalisation of [12]] Let $\mathcal{C}$ be a seminorm and $\mathcal{I}$ an edge-implicator on RGB. Consider a colour image A and a 'normalized' colour structuring element B, that is, $(\forall y \in \mathbb{R}^2)(\exists z \in \mathbb{R}^2)(B(z-y) = \mathbf{1})$. It holds that*

$$\vec{E}_{\mathcal{I}}(A,B) \subseteq_{RGB} \vec{D}_{\mathcal{C}}(A,B).$$

Property 4.64. *[Generalisation of [12]] Let $\mathcal{C}$ be a seminorm and $\mathcal{I}$ an edge-implicator on RGB. For every colour image A and every colour structuring element B, it holds that*

$$B(\mathbf{0}) =_{RGB} \mathbf{1} \quad \Rightarrow \quad A \subseteq_{RGB} \vec{D}_{\mathcal{C}}(A,B) \text{ and } \vec{E}_{\mathcal{I}}(A,B) \subseteq_{RGB} A.$$

Property 4.65 (Interaction with intersection and union). *[Generalisation of [12]] Consider a family $(A_i)_{i=1}^n$ of colour images and a family $(B_i)_{i=1}^n$ of colour structuring elements. For the $\mathcal{C}$-'colour' dilation it holds that*

$$\begin{aligned} \vec{D_{\mathcal{C}}}(\bigcap_{i=1}^{n} A_i, B) &\subseteq_{RGB} \bigcap_{i=1}^{n} \vec{D_{\mathcal{C}}}(A_i,B) \\ \vec{D_{\mathcal{C}}}(A, \bigcap_{i=1}^{n} B_i) &\subseteq_{RGB} \bigcap_{i=1}^{n} \vec{D_{\mathcal{C}}}(A,B_i); \\ \vec{D_{\mathcal{C}}}(\bigcup_{i=1}^{n} A_i, B) &=_{RGB} \bigcup_{i=1}^{n} \vec{D_{\mathcal{C}}}(A_i,B) \\ \vec{D_{\mathcal{C}}}(A, \bigcup_{i=1}^{n} B_i) &=_{RGB} \bigcup_{i=1}^{n} \vec{D_{\mathcal{C}}}(A,B_i). \end{aligned}$$

For the $\mathcal{I}$-'colour' erosion it holds that

$$\begin{aligned}
\vec{E}_{\mathcal{I}}(\bigcap_{i=1}^{n} A_i, B) &=_{RGB} \bigcap_{i=1}^{n} \vec{E}_{\mathcal{I}}(A_i, B) \\
\vec{E}_{\mathcal{I}}(A, \bigcap_{i=1}^{n} B_i) &\supseteq_{RGB} \bigcup_{i=1}^{n} \vec{E}_{\mathcal{I}}(A, B_i); \\
\vec{E}_{\mathcal{I}}(\bigcup_{i=1}^{n} A_i, B) &\supseteq_{RGB} \bigcup_{i=1}^{n} \vec{E}_{\mathcal{I}}(A_i, B) \\
\vec{E}_{\mathcal{I}}(A, \bigcup_{i=1}^{n} B_i) &=_{RGB} \bigcap_{i=1}^{n} \vec{E}_{\mathcal{I}}(A, B_i).
\end{aligned}$$

Some examples of conjunctors $\mathcal{C}$ on $(RGB, \leq_{RGB})$ are

1. $\mathcal{C}_{\min}(\gamma, \delta) = \min_{RGB}(\gamma, \delta)$
2. $\mathcal{C}_*(\gamma, \delta) \quad = \gamma * \delta, \qquad\qquad \forall \gamma, \delta \in RGB.$

One can easily show that the conjunctor $\mathcal{C}_{\min}$ as well as the conjunctor $\mathcal{C}_*$ is a t-norm on $(RGB, \leq_{RGB})$; we note $\mathcal{T}_{\min}$ and $\mathcal{T}_*$. The $\mathcal{S}$-implicators induced by $\mathcal{T}_{\min}$ (and $\mathcal{T}_*$) and the standard negator $\mathcal{N}_s(c) = \mathbf{1}_{RGB} - c$, for all $c \in RGB$, on $(RGB, \leq_{RGB})$ are then given by

1. $\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}(\gamma, \delta) = \max_{RGB}(\mathbf{1} - \gamma, \delta)$
2. $\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}(\gamma, \delta) \quad = \mathbf{1} - (\gamma * (\mathbf{1} - \delta)), \qquad \forall \gamma, \delta \in RGB.$

• Umbra approach

Definition 4.66. *Let A be a colour image and B a colour structuring element (both represented as $\mathbb{R}^2 - (RGB, \leq_{RGB})$ mappings). The **umbra 'colour' dilation** $\vec{D}_u(A, B)$ and the **umbra 'colour' erosion** $\vec{E}_u(A, B)$ are the colour images given by*

$$\begin{aligned}
\vec{D}_u(A, B)(y) &\stackrel{def}{=}_{RGB} \max_{x \in T_y(d_B)} A(x) \oplus B(x - y) \qquad \textit{for } y \in \mathbb{R}^2, \\
\vec{E}_u(A, B)(y) &\stackrel{def}{=}_{RGB} \min_{x \in T_y(d_B)} A(x) \ominus B(x - y) \qquad \textit{for } y \in \mathbb{R}^2,
\end{aligned}$$

where $\oplus$ and $\ominus$ are colour mix operators.

Definition 4.67. *Let c and c' be two colours in RGB. We define the colour mix operators $\oplus$ and $\ominus$ for c and c' as*

$$\begin{aligned}
c \oplus c' &=_{RGB} \begin{cases} \mathbf{0}_{RGB} & \textit{if } c =_{RGB} \mathbf{0}_{RGB} \\ c +_{RGB} c' & \textit{otherwise} \end{cases}, \\
c \ominus c' &=_{RGB} \begin{cases} \mathbf{1}_{RGB} & \textit{if } c =_{RGB} \mathbf{1}_{RGB} \\ c -_{RGB} c', & \textit{otherwise} \end{cases}.
\end{aligned}$$

Property 4.68.

$$\vec{D_u}(\overline{\mathbf{0}}, B) =_{RGB} \overline{\mathbf{0}} \quad \textit{and} \quad \vec{E_u}(\overline{\mathbf{1}}, B) =_{RGB} \overline{\mathbf{1}}.$$

Proof
For all y in $\mathbb{R}^2$ it holds that

$$\vec{D_u}(\overline{\mathbf{0}}, B)(y) = \max_{x \in T_y(B)} \overline{\mathbf{0}}(x) \ \oplus \ B(x-y) = \max_{x \in T_y(B)} \mathbf{0} \ \oplus \ B(x-y) = \mathbf{0} = \overline{\mathbf{0}}(y),$$

$$\vec{E_u}(\overline{\mathbf{1}}, B)(y) = \min_{x \in T_y(B)} \overline{\mathbf{1}}(x) \ \ominus \ B(x-y) = \min_{x \in T_y(B)} \mathbf{1} \ \ominus \ B(x-y) = \mathbf{1} = \overline{\mathbf{1}}(y).$$

■

Property 4.69 (Duality dilation-erosion).

$$\begin{aligned} \vec{D_u}(A, B) &=_{RGB} co(\vec{E_u}(co(A), B)) \\ \vec{E_u}(A, B) &=_{RGB} co(\vec{D_u}(co(A), B)). \end{aligned}$$

Proof
1) *If $A \neq_{RGB} \overline{\mathbf{1}}$ we get:*

$$\begin{aligned} co(\vec{D_u}(co(A), B))(y) &= \mathbf{1}_{RGB} - \vec{D_u}(\mathbf{1}_{RGB} - A, B)(y) \\ &= \mathbf{1}_{RGB} - \max_{x \in T_y(d_B)} (\mathbf{1}_{RGB} - A)(x) \oplus B(x-y) \\ &= \min_{x \in T_y(d_B)} \mathbf{1}_{RGB} - ((\mathbf{1}_{RGB} - A)(x) \oplus B(x-y)) \\ &= \min_{x \in T_y(d_B)} \mathbf{1}_{RGB} - \Big(\frac{1 - r_{A(x)} + r_{B(x-y)}}{2}, \\ &\quad \frac{1 - g_{A(x)} + g_{B(x-y)}}{2}, \frac{1 - b_{A(x)} + b_{B(x-y)}}{2} \Big) \\ &= \min_{x \in T_y(d_B)} \Big(\frac{r_{A(x)} + 1 - r_{B(x-y)}}{2}, \\ &\quad \frac{g_{A(x)} + 1 - g_{B(x-y)}}{2}, \frac{b_{A(x)} + 1 - b_{B(x-y)}}{2} \Big) \\ &= \vec{E_u}(A, B)(y), \ \forall y \in \mathbb{R}^2. \end{aligned}$$

2) *If* $A \neq_{RGB} \overline{\mathbf{0}}$ *we get:*

$$\begin{aligned}
co(\vec{E_u}(co(A),B))(y) &= \mathbf{1}_{RGB} - \vec{E_u}(\mathbf{1}_{RGB} - A, B)(y) \\
&= \mathbf{1}_{RGB} - \min_{x \in T_y(d_B)} (\mathbf{1}_{RGB} - A)(x) \ominus B(x-y) \\
&= \max_{x \in T_y(d_B)} \mathbf{1}_{RGB} - ((\mathbf{1}_{RGB} - A)(x) \oplus \\
&\quad (\mathbf{1}_{RGB} - B)(x-y)) \\
&= \max_{x \in T_y(d_B)} \mathbf{1}_{RGB} - \left(\frac{1 - r_{A(x)} + 1 - r_{B(x-y)}}{2}, \right. \\
&\quad \left. \frac{1 - g_{A(x)} + 1 - g_{B(x-y)}}{2}, \frac{1 - b_{A(x)} + 1 - b_{B(x-y)}}{2} \right) \\
&= \max_{x \in T_y(d_B)} \left(\frac{r_{A(x)} + r_{B(x-y)}}{2}, \right. \\
&\quad \left. \frac{g_{A(x)} + g_{B(x-y)}}{2}, \frac{b_{A(x)} + b_{B(x-y)}}{2} \right) \\
&= \vec{D_u}(A,B)(y), \ \forall y \in \mathbb{R}^2.
\end{aligned}$$

3) *If* $A =_{RGB} \overline{\mathbf{1}}$ *we get:*

$$\begin{aligned}
co(\vec{D_u}(co(\overline{\mathbf{1}}),B))(y) &= \mathbf{1}_{RGB} - \vec{D_u}(\overline{\mathbf{0}}, B)(y) \\
&= \mathbf{1}_{RGB} - \mathbf{0}_{RGB} \\
&= \mathbf{1}_{RGB} \\
&= \vec{E_u}(\overline{\mathbf{1}}, B)(y), \ \forall y \in \mathbb{R}^2.
\end{aligned}$$

4) *If* $A =_{RGB} \overline{\mathbf{0}}$ *we get:*

$$\begin{aligned}
co(\vec{E_u}(co(\overline{\mathbf{0}}),B))(y) &= \mathbf{1}_{RGB} - \vec{E_u}(\overline{\mathbf{1}}, B)(y) \\
&= \mathbf{1}_{RGB} - \mathbf{1}_{RGB} \\
&= \mathbf{0}_{RGB} \\
&= \vec{D_u}(\overline{\mathbf{0}}, B)(y), \ \forall y \in \mathbb{R}^2.
\end{aligned}$$

■

4.3.9 Experimental Results

In figures 4.26 to 4.43 you see the results of the different kinds of morphological operators in RGB. We have used the same structuring elements as before. In figure 4.26, 4.27 and 4.28 the t-morphological colour operators dilation and erosion in RGB are illustrated for the component-based and the proposed approach. The results look very

Figure 4.26: T-morphological operators in RGB: the component-based t-dilation $D_t(C, B')$ and the t-'colour' dilation $\vec{D}_t(C, B')$.

Figure 4.27: T-morphological operators in RGB: the component-based t-erosion $E_t(C, B')$ and the t-'colour' erosion $\vec{E}_t(C, B')$.

similar, but pay attention to the hat for example. Figure 4.28 shows that more colours presented in the original colour image occur in the results with our approach than in the results with the other approach. This way more details from the original image are preserved with our method.

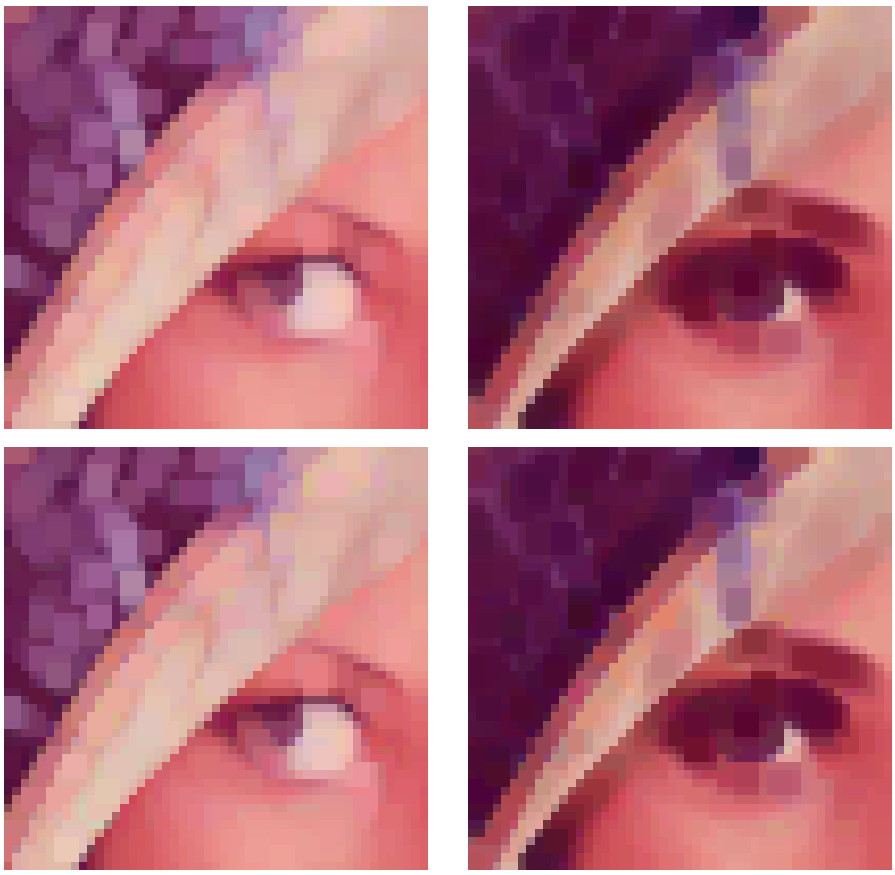


Figure 4.28: T-morphological operators in RGB: left column: the component-based t-dilation $D_t(C, B')$ and the t-'colour' dilation $\vec{D}_t(C, B')$, right column: the component-based t-erosion $E_t(C, B')$ and t-'colour' erosion $\vec{E}_t(C, B')$.

Figure 4.29, 4.30 and 4.31 show the results for the fuzzy morphological colour operators in RGB with conjunctor-implicator pair $(\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min},\mathcal{N}_s})$ for the component-based approach and our method. Again, the obtained results are very similar, but some details, especially along edges, are better visible with the proposed method.

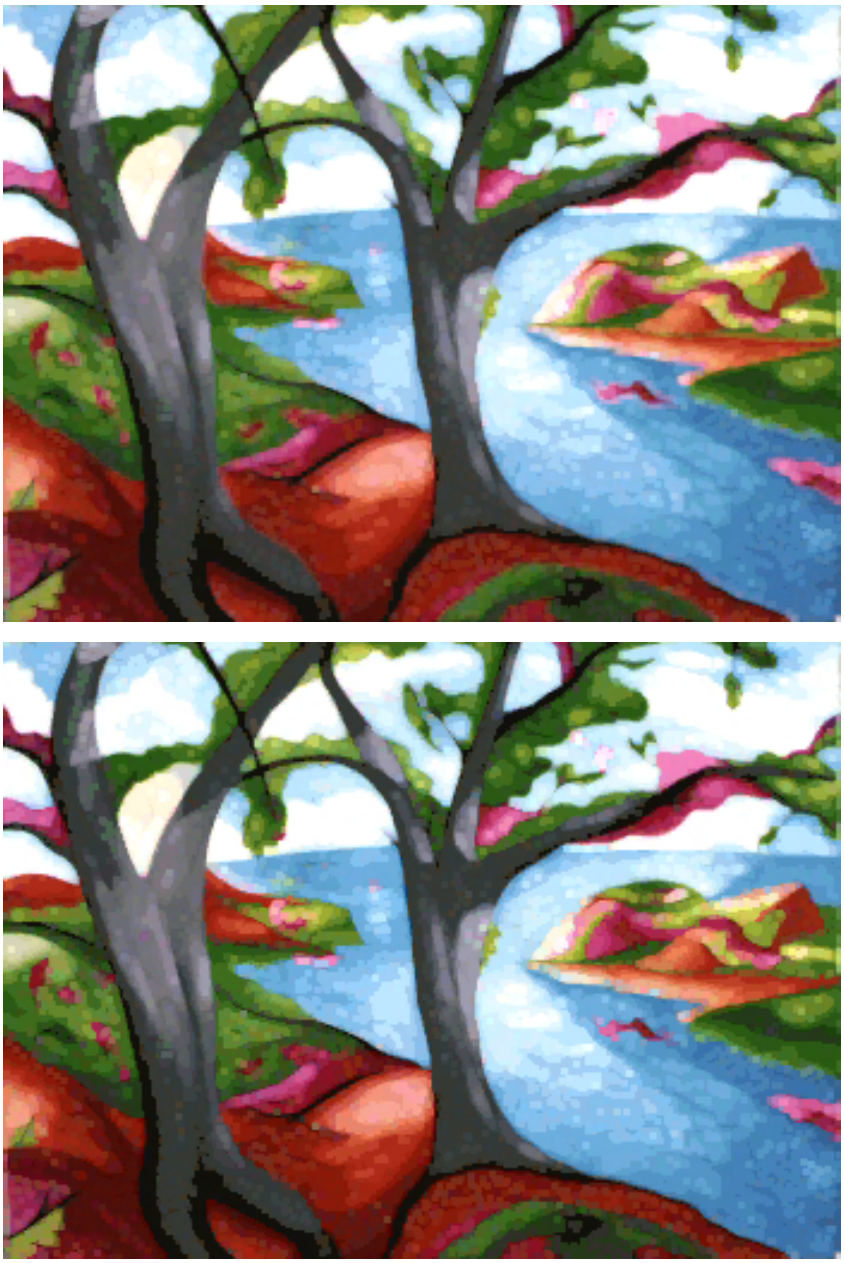


Figure 4.29: Fuzzy morphological operators for $(\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}}, \mathcal{N}_s)$ in RGB: the component-based fuzzy dilation $D_{\mathcal{T}_{\min}}(C, B^{Wh})$ and the fuzzy 'colour' dilation $\vec{D}_{\mathcal{T}_{\min}}(C, B^{Wh})$.

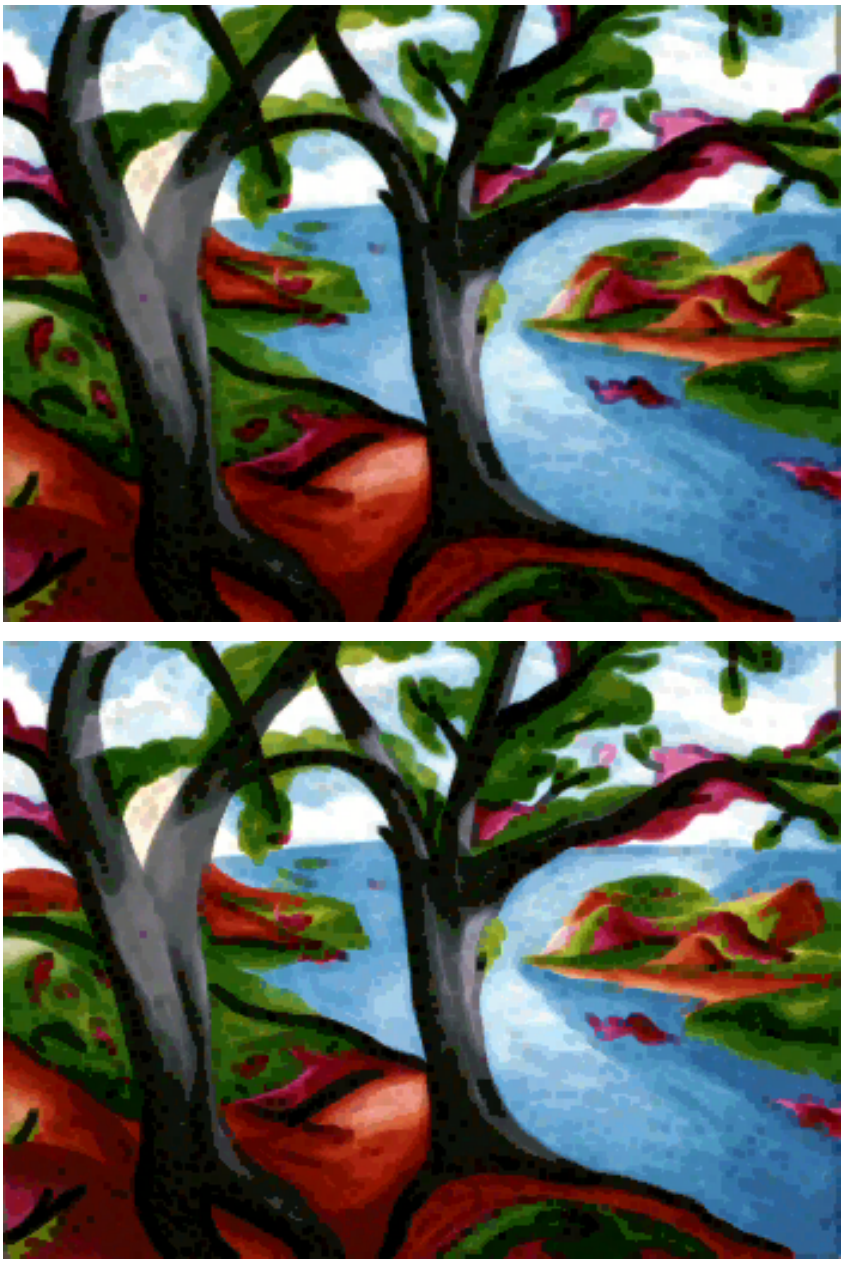


Figure 4.30: Fuzzy morphological operators for $(\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s})$ in RGB: the component-based fuzzy erosion $E_{\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}}(C, B^{Wh})$ and the fuzzy 'colour' erosion $\vec{E}_{\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}}(C, B^{Wh})$.

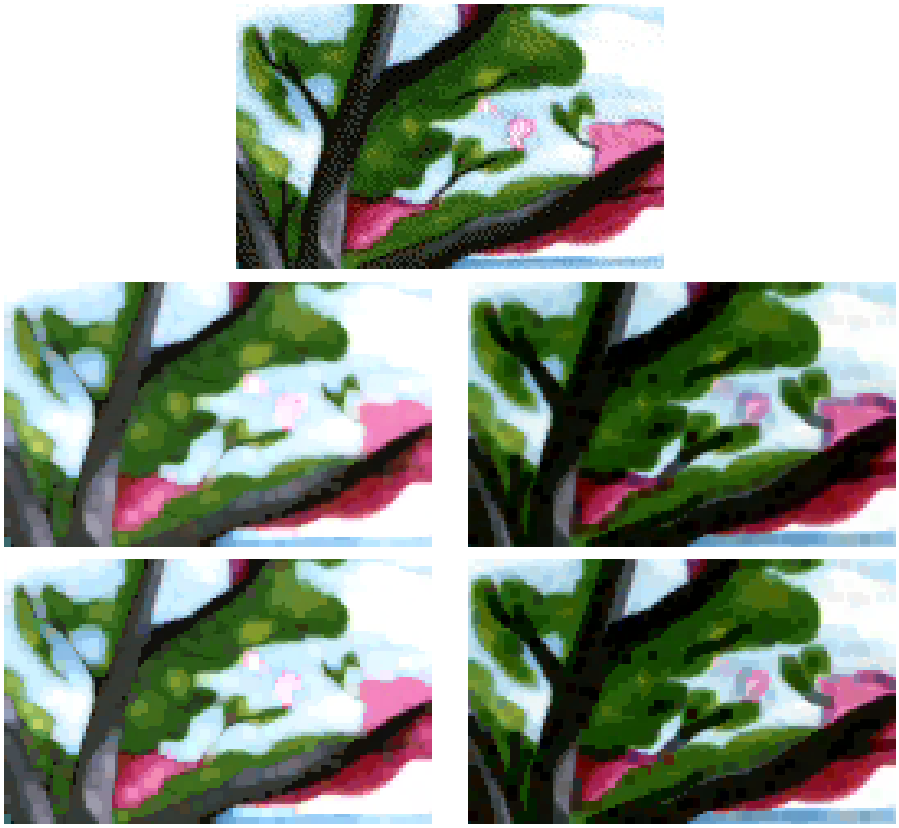


Figure 4.31: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_{\min}, \mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s})$ in RGB: at the top: original image C, left column: the component-based fuzzy dilation $D_{\mathcal{T}_{\min}}(C, B^{Wh})$ and the fuzzy 'colour' dilation $\vec{D}_{\mathcal{T}_{\min}}(C, B^{Wh})$, right column: the component-based fuzzy erosion $E_{\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}}(C, B^{Wh})$ and the fuzzy 'colour' erosion $\vec{E}_{\mathcal{I}_{\mathcal{T}_{\min}, \mathcal{N}_s}}(C, B^{Wh})$.

The fuzzy colour dilation and erosion for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in RGB for the component-based approach and the proposed method are shown in figures 4.32, 4.33 and 4.34. The results for both approaches are quite similar.

Figure 4.32: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_* \overset{\cdot}{\cdot} \mathcal{N}_s})$ in RGB: the component-based fuzzy dilation $D_{\mathcal{T}_*}(C, B'')$ and the fuzzy 'colour' dilation $\vec{D}_{\mathcal{T}_*}(C, B'')$.

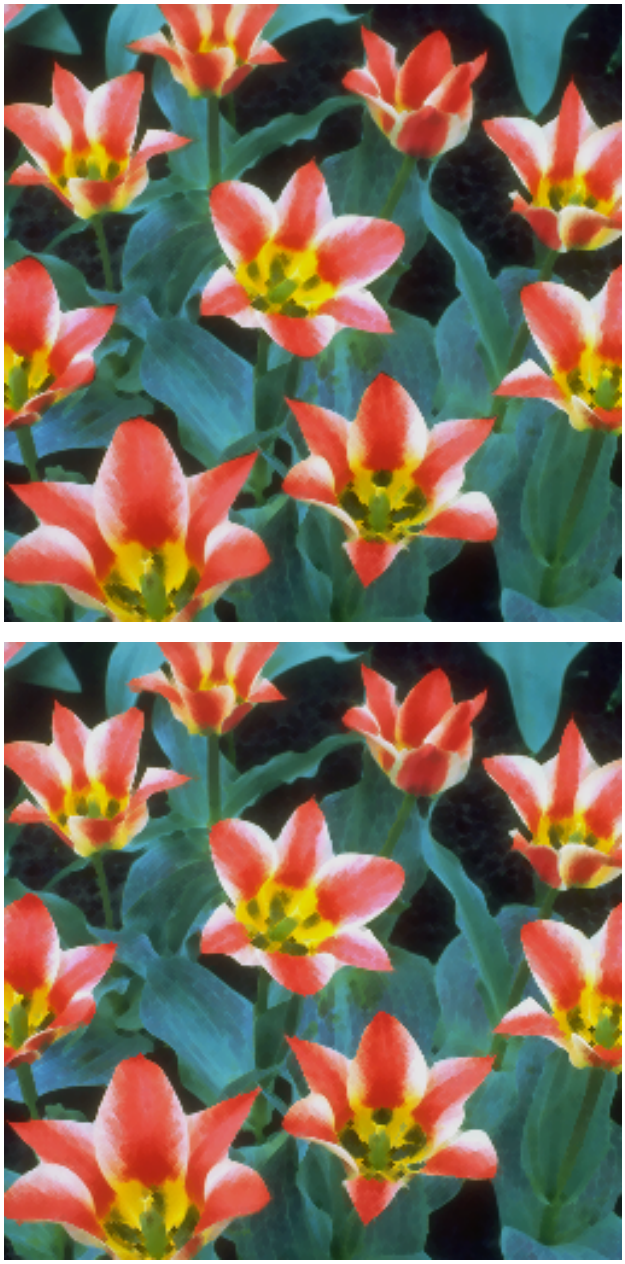


Figure 4.33: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in RGB: the component-based fuzzy erosion $E_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$ and the fuzzy 'colour' erosion $\vec{E}_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$.

Figure 4.34: Fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_*, \mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s})$ in RGB: at the top: the component-based fuzzy dilation $D_{\mathcal{T}_*}(C, B'')$ and the fuzzy 'colour' dilation $\vec{D}_{\mathcal{T}_*}(C, B'')$, at the bottom: the component-based fuzzy erosion $E_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$ and the fuzzy 'colour' erosion $\vec{E}_{\mathcal{I}_{\mathcal{T}_*, \mathcal{N}_s}}(C, B'')$.

The umbra colour operators in RGB for the component-based approach and the proposed method are illustrated in figures 4.35, 4.36 and 4.37. The original results as well as the results where the new colours have been replaced by the original colours of the corresponding pixel positions are shown.

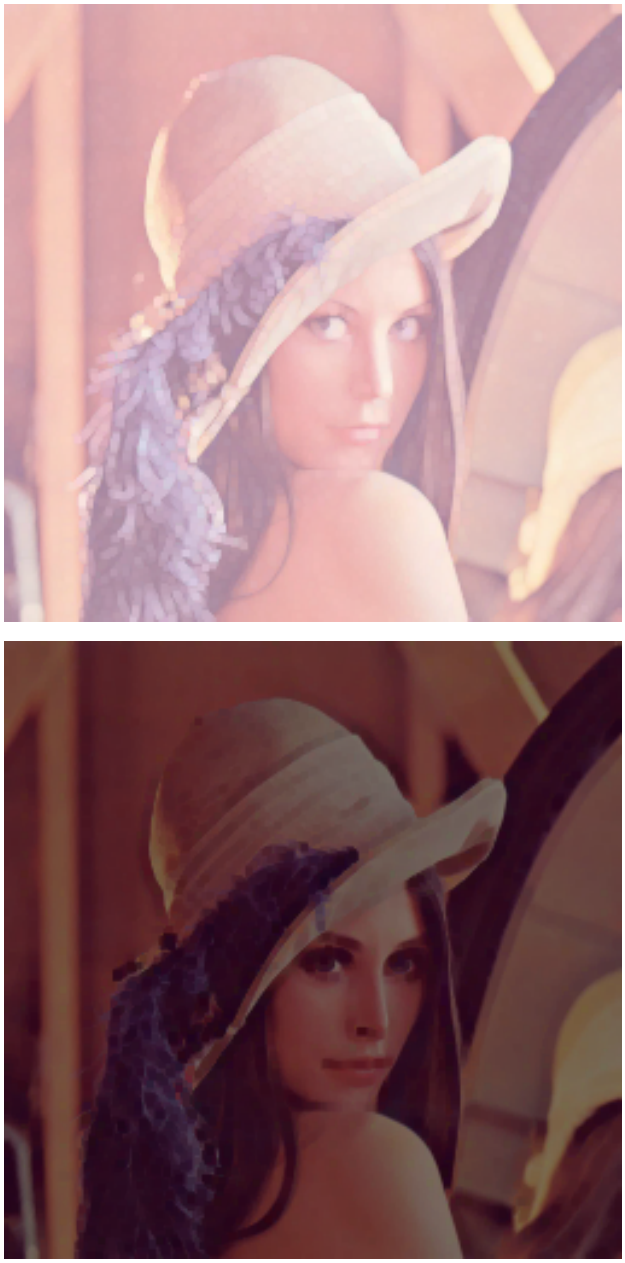


Figure 4.35: U-morphological operators in RGB: the u-'colour' dilation $\vec{D}_u(C, B^{Wh})$ (top) and the u-'colour' erosion $\vec{E}_u(C, B^{Wh})$ (bottom).

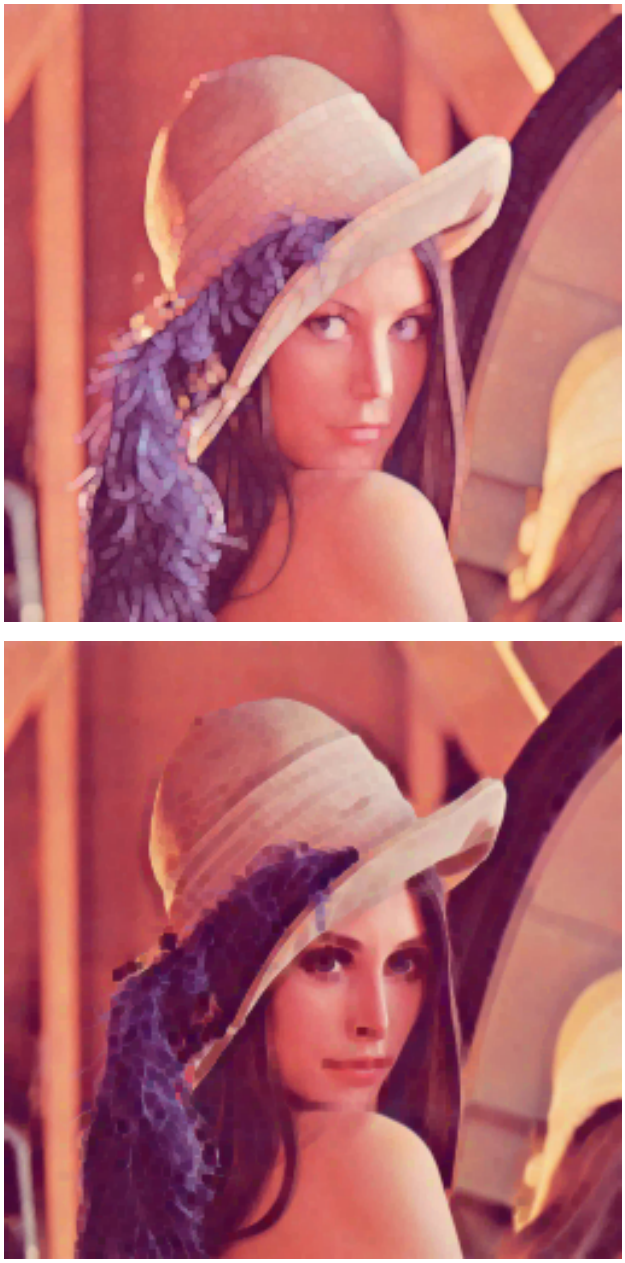


Figure 4.36: U-morphological operators in RGB: the u-'colour' dilation $\vec{D}_u(C, B^{Wh})$ (top) and the u-'colour' erosion $\vec{E}_u(C, B^{Wh})$ (bottom) containing no new colours.

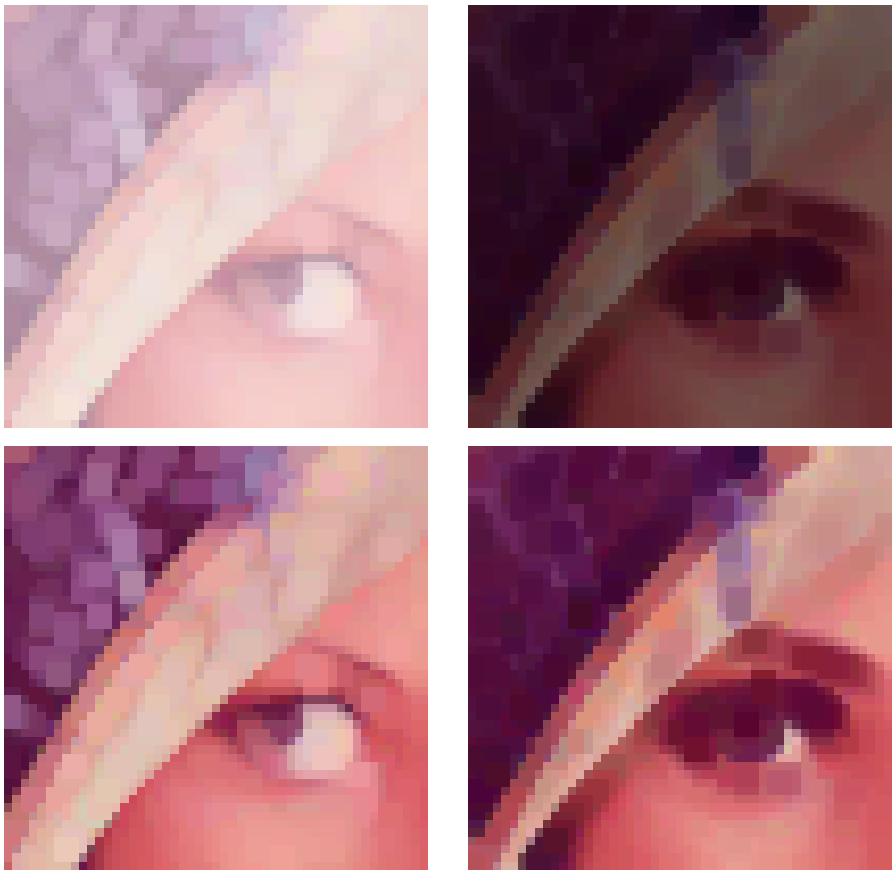


Figure 4.37: U-morphological operators in RGB: left column: the u-'colour' dilation $\vec{D}_u(C, B^{Wh})$, right column: the u-'colour' erosion $\vec{E}_u(C, B^{Wh})$, from top to bottom: containing new colours and containing no new colours.

We have compared our technique in RGB with the following state-of-the-art methods: we have used the α-modulus lexicographical order $I - RGB_\alpha$ presented in [5] to determine the t-morphological operators dilation and erosion and the proposed vector dilation and vector erosion of [8] using a reduced ordering with as measurement functions the luminance image and the Euclidean distance.

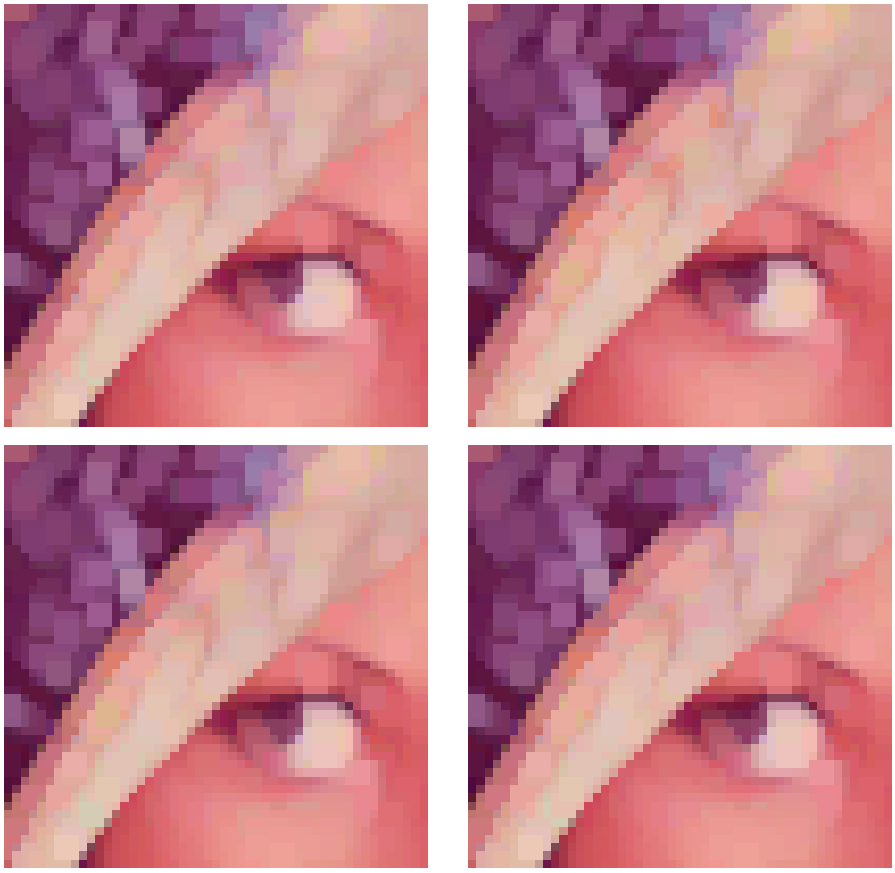


Figure 4.38: The t-dilation in RGB by structuring element B': above: using our RGB ordering (left) and the $I-RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

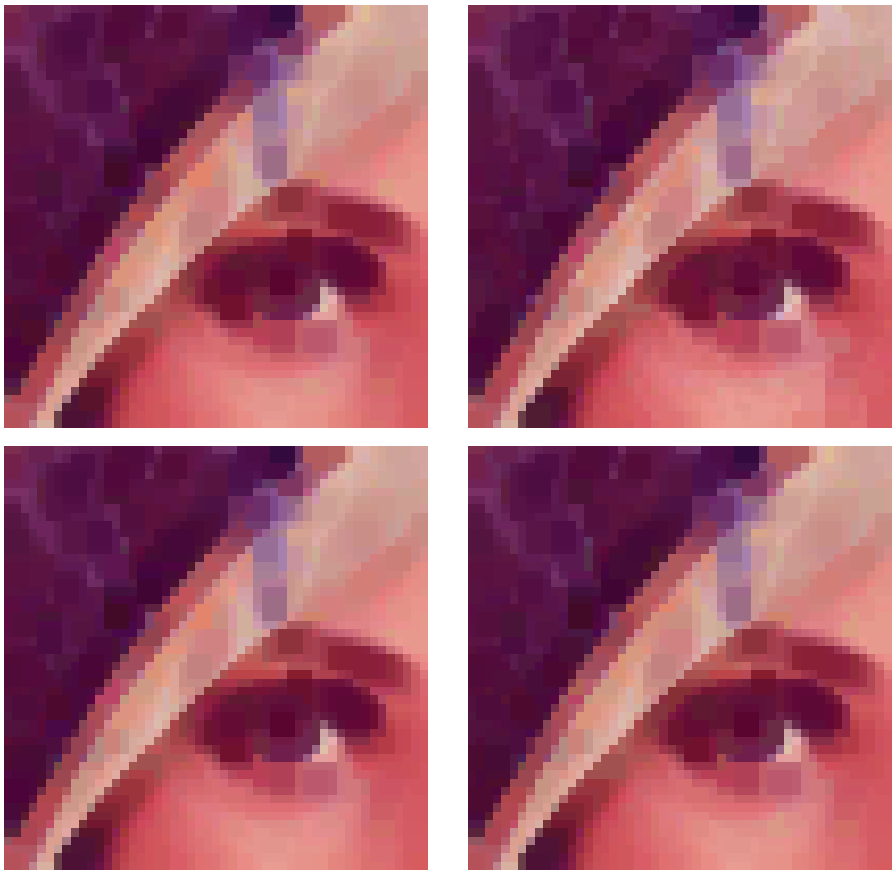


Figure 4.39: The t-erosion in RGB by structuring element B': above: using our RGB ordering (left) and the $I-RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

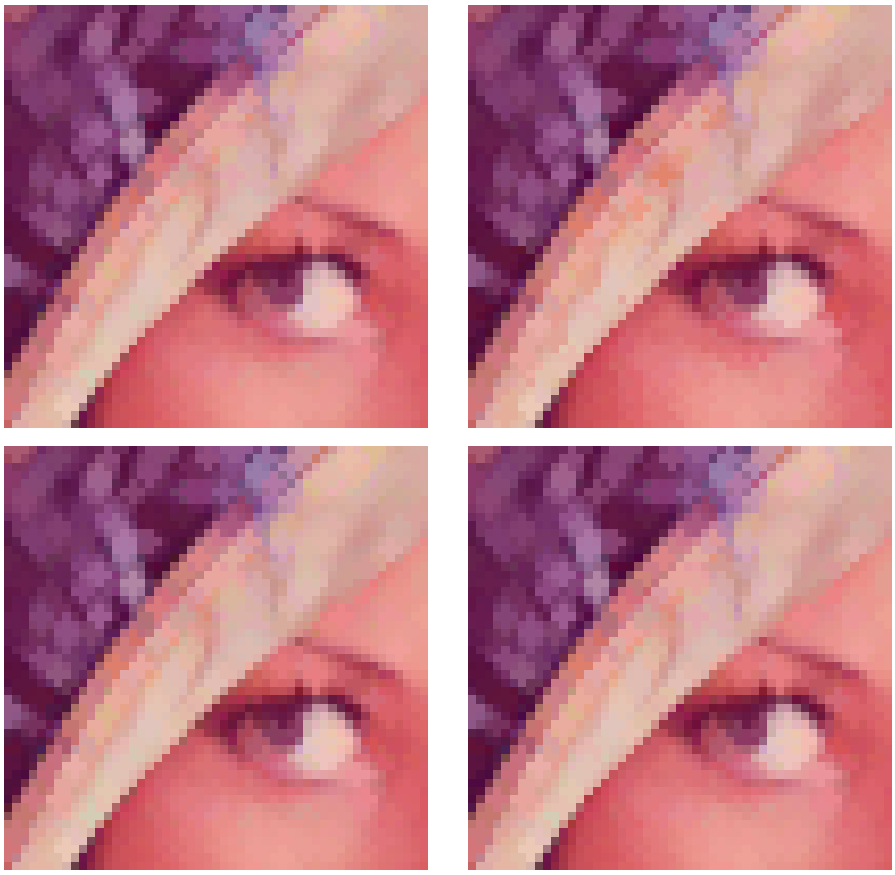


Figure 4.40: The t-dilation in RGB by structuring element B''_{RGB}: above: using our RGB ordering (left) and the $I - RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

Figure 4.41: The t-erosion in RGB by structuring element B''_{RGB}: above: using our RGB ordering (left) and the $I - RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

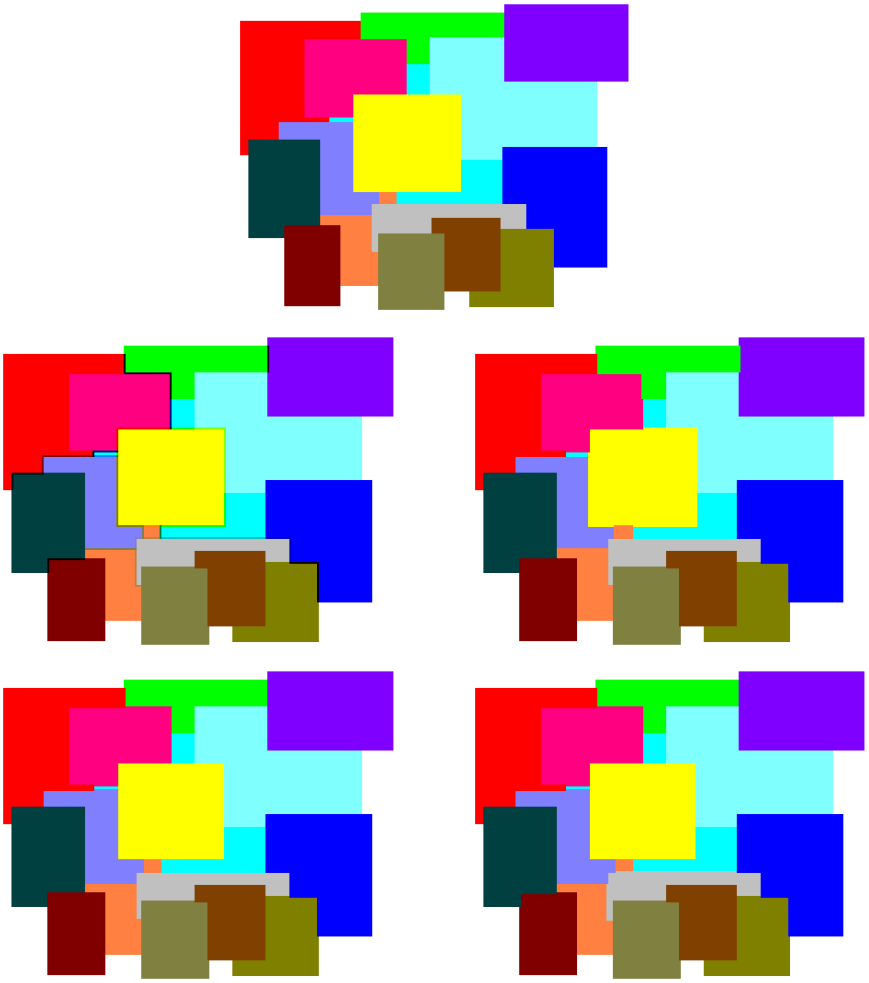


Figure 4.42: The t-dilation in RGB by structuring element B': above: the original colour image, in the middle: using our RGB ordering (left) and the $I - RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

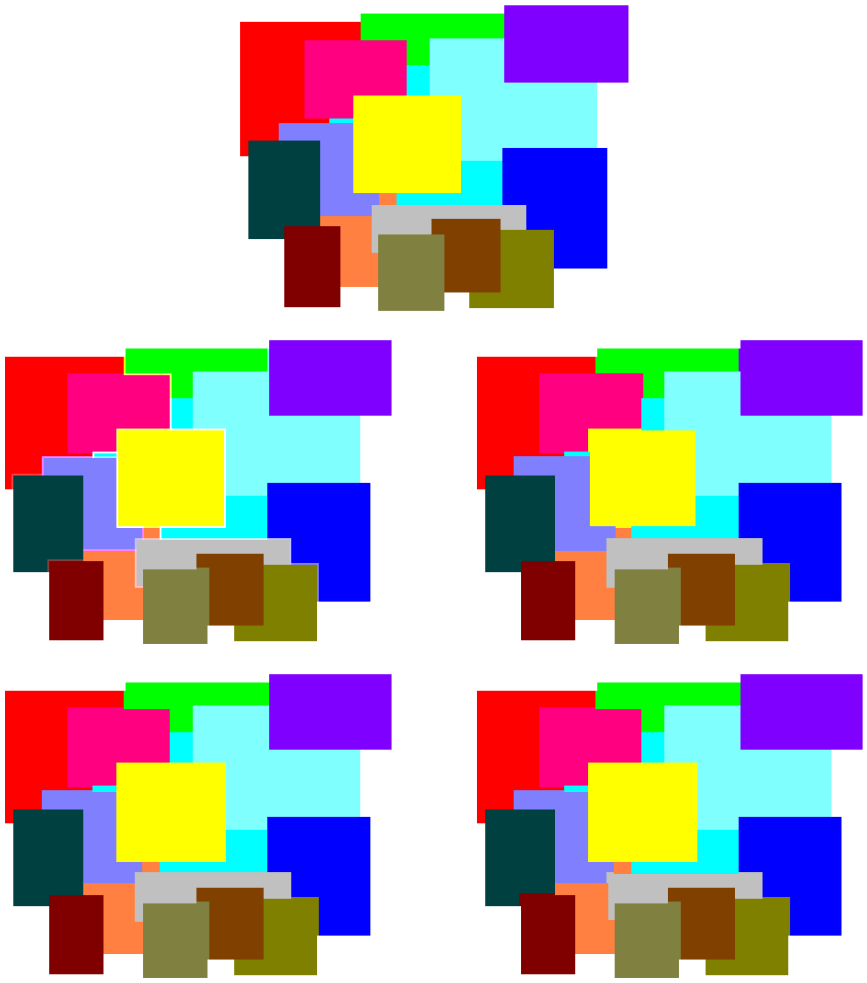


Figure 4.43: The t-erosion in RGB by structuring element B': above: the original colour image, in the middle: using our RGB ordering (left) and the $I - RGB_{\alpha=10}$ ordering (right), below: using a reduced ordering with the luminance image (left) and the Euclidean distance (right) as measurement functions.

All methods give similar results, but look carefully at the hat of the Lena image. More details from the original image are visible because more colours from the original colour image are presented in the results with our approach than in the results obtained by the other approaches.

At last we give an overview of the properties that still hold for the new colour morphological operators in the RGB, HSV and L*a*b* colour model:

	Threshold	**Fuzzy**	**Umbra**
Duality dilation-erosion	×	×	×
Monotonicity	×	×	
Inclusion	×	×	
Interaction with intersection and union	×	×	

4.3.10 Conclusion

We have presented a new vector ordering procedure for morphological processing of colour images on fuzzy sets, umbra and thresholding techniques. The problem of looking for a vector ordering for colour or multivariate morphological image processing is not new and is being developed since the early 90's. What is new here is the used approach, namely through the umbra approach and fuzzy set theory. We may conclude that our new method provides better results than those obtained by the component-based approach and similar or better results than those obtained by other state-of-the-art methods. On the one hand one great advantage is that the existing correlations between the different colour components are taken into account. Firstly, the colours are preserved and thus no new colours appear after applying the new vector t- and fuzzy morphological operators for $(\mathcal{C}, \mathcal{I}) = (\mathcal{T}_{min}, \mathcal{I}_{\mathcal{T}_{min}, \mathcal{N}_s})$ to colour images. Secondly, more details from the original colour image are preserved and thus visible. On the other hand visual inspection shows that there still may appear some little artefacts. As future work we can set up an experiment regarding the psycho visual behaviour of similarity measures, which can be useful for the evaluation of morphological operators.

Chapter 5

Image Magnification

We now introduce a new image magnification approach [14]. First we demonstrate the hit-or-miss transformation and explain the pixel replication or nearest neighbour interpolation, used as the first 'trivial' interpolation step in our method. Next we discuss our corner detection method, using different kinds of structuring elements, and describe our corner correction, first for magnification by a factor 2 and then for magnification by an integer factor $n > 2$. In the previous chapter we have presented a new vector ordering $\leq_{RGB}$ for colours modelled in the RGB colour model. We have also defined a complement co for colours in RGB, with which our new ordering is compatible. Here we use $\leq_{RGB}$ in our morphological magnification method, where we need the compatibility of $\leq_{RGB}$ with co to detect corners in an image by the hit-or-miss transformation. Thereafter we compare our magnification method experimentally to other well-known approaches. The results show that our method provides a visual improvement in quality on existing techniques: almost all jagged effects have been removed so that the edges become smooth. Finally, in the last section, we present the extension of our new magnification method towards colour images in RGB with 'vague' edges.

5.1 The Hit-or-Miss Transformation

Consider a binary image X and two binary structuring elements A and B.
The **hit-or-miss operator of** X **by** A **and** B is defined as

$$X \otimes (A, B) = E(X, A) \cap E(co(X), B),$$

where $co(X)$ is the complement of X w.r.t. $\mathbb{R}^2$. The result is empty if $A \cap B \neq \emptyset$. The name hit-or-miss operator can be explained as follows: a pixel h belongs to the hit-or-miss transformation $X \otimes (A, B)$ if and only if $T_h(A)$ does not hit (intersect with) $co(X)$ and $T_h(B)$ does not hit X. The hit-or-miss operator is very useful for the detection of points inside an image with certain (local) geometric properties, e.g. isolated

points, edge points, corner points.

The basic idea behind the hit-or-miss transformation consists in extracting image pixels having a given neighbouring configuration. The neighbouring configuration is therefore defined by two disjoint sets, the first for the object pixels and the second for the background pixels. These two sets are two composite structuring elements that have the same origin. As an example we show the detection of the upper-left corner points of objects in an image in figure 5.1. The structuring elements A and B have been chosen in such a way that the hit-or-miss operator detects the lower left corner points of the original image.

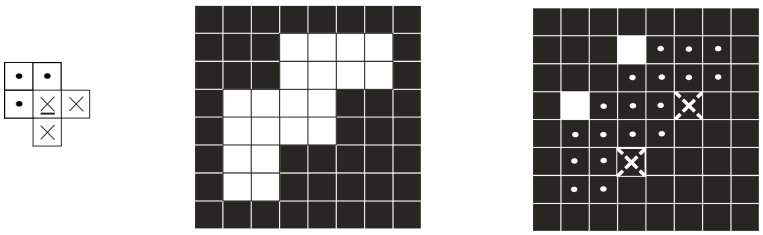


Figure 5.1: From left to right: The structuring elements (A, B), where A contains the white pixels with ($\times$)-symbol and B the white pixels with ($\cdot$)-symbol, the original binary image X and the hit-or-miss transformation $X \otimes (A, B)$ (only the white pixels).

More information about the hit-or-miss transformation can be found in [25].

5.2 New Morphological Image Interpolation Method to Magnify Images with Sharp Edges

5.2.1 Pixel Replication or Nearest Neighbour Interpolation

When we magnify an image V times, the number of pixels will increase (V^2 times). The easiest way to enlarge an image is to copy the existing pixel values to the new neighbouring pixels. If we magnify an image V times, one pixel in the original image will be replaced by a square of $V \times V$ pixels in the new image. This is called **pixel replication** or **nearest neighbour interpolation**. The result is quite poor, but we can use it as a first 'trivial' interpolation step.

5.2.2 Corner Detection

To detect the unwanted jaggies in the nearest neighbour interpolated image, we first determine the inner edge-image of the blown up image using the structuring element

B^{Wh}_{RGB} and then apply the hit-or-miss operator to obtain the positions of the object corner edge pixels. The advantage of the internal morphological gradient is that this gradient gives the correct positions of corner pixels in an image. Let us call O the nearest neighbour interpolated image of an original image X, $co(O)$ the complement of O, O_{edge} the inner edge-image of O, and O^c_{edge} the inner edge-image of $co(O)$. If the original image X is a binary image, the blown up image O will be a binary image, and so will the inner edge-image O_{edge}. On the other hand, if the original image X is a colour image, the inner edge-image O_{edge} of the blown up image O will also be a colour image, but we transform it into a greyscale image by giving all non-black pixels a grey value:

$O_{edge} = double(O_{edge})$;
$[mO_{edge}, nO_{edge}, 3] = \text{size}(O_{edge})$;
$O_{edge,new} = \text{zeros}(mO_{edge}, nO_{edge}, 3)$;

for $i = 1 : mO_{edge}$
  for $j = 1 : nO_{edge}$
    $O_{edge,new}(i, j, 1) = \max(\max(O_{edge}(i, j, 1), O_{edge}(i, j, 2)), O_{edge}(i, j, 3))$;
    $O_{edge,new}(i, j, 2) = O_{edge,new}(i, j, 1)$;
    $O_{edge,new}(i, j, 3) = O_{edge,new}(i, j, 1)$;
  end;
end;
$O_{edge} = \text{uint8}(O_{edge,new})$;

Analogously for the inner edge-image O^c_{edge} of the complementary image $co(O)$.

With a given pair of structuring elements (A, B) we first determine the hit-or-miss transformation $O_{edge} \otimes (A, B)$ and secondly the hit-or-miss transformation $O^c_{edge} \otimes (A, B)$. This way we will not only detect corners of objects in O, but also corners of objects in $co(O)$. Not all the corner pixels in the image should be changed, because some corners are 'real' corners, which have to be preserved in the magnified image, whereas others are part of jaggies and have to be removed. Therefore we will use different kinds of binary structuring elements in our hit-or-miss transformation. The structuring elements used for magnification by a factor 2 are shown in figure 5.2, where A_i contains the white pixels $(\times)$ and B_i the white pixels $(\cdot)$, $i = 1 \ldots 9$. The other structuring elements are rotated or reflected versions of these. For example, the structuring elements (A_1, B_1) will allow us to detect all upper-left corner pixels, while using structuring elements that are properly rotated versions of (A_1, B_1), we will detect all upper-right, lower-left and lower-right corners. And we not only look for corners of the 'foreground' objects, but also for corners of the 'background' objects. In the example in section 5.1 (figure 5.1), not only the white pixels ($X \otimes (A, B)$) will be detected (foreground object corners), but also the black pixels with white $\times$-symbol ($co(X) \otimes (A, B)$) (background object corners). Consequently we have 8 different kinds

of corner pixels (4 ‘foreground’ object corner pixels and 4 ‘background’ object corner pixels).

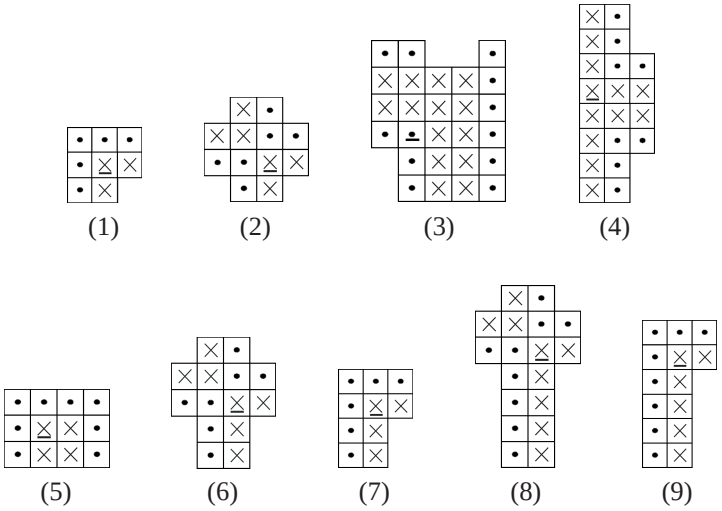


Figure 5.2: The used binary structuring elements (1) (A_1, B_1) ... (9) (A_9, B_9) (the underlined element corresponds to the origin of coordinates) for magnification by a factor 2.

5.2.3 Corner Correction

In this section we explain our proposed corner transformation method, which is a trade-off between blur and jaggies, for magnification by a factor 2.

Binary images

If the original image X is a binary image, then the nearest neighbour interpolated image O of X is also a binary image, and so are the complement $co(O)$ of O, the inner edge-image O_{edge} of O and the inner edge-image O^c_{edge} of $co(O)$. We can apply the hit-or-miss operator on the binary images O_{edge} and O^c_{edge}, and get binary images as result.

Step 1. We look for corners determined by the structuring elements (A_1, B_1) and their rotations. For example, we detect an upper-left foreground corner pixel a or an upper-left background corner pixel a^* at position (i, j) in O (see figure 5.3). An upper-left foreground corner pixel will be determined by the hit-or-miss transformation $O_{edge} \otimes (A_1, B_1)$, while an upper-left background corner pixel will be determined by the hit-or-miss transformation $O^c_{edge} \otimes (A_1, B_1)$. Then we change the colour of the

pixel at position (i, j) in the blown up image O by a mixture of the 'foreground' and 'background' colour of the surrounding pixels. We obtain this by adding a mixture of the pixel values at positions $(i-1, j)$ and $(i, j-1)$ to the pixel value at position (i, j). Let $[r_c, g_c, b_c]$ and $[r_{c'}, g_{c'}, b_{c'}]$ be the RGB colour vectors of the pixels $O(i-1, j)$ and $O(i, j-1)$ respectively, and if we call our new image O_{new}, then $O_{new}(i, j)(r, g, b) = (O(i-1, j)(r_c, g_c, b_c) + O(i, j-1)(r_{c'}, g_{c'}, b_{c'})) + O(i, j)(r_i, g_i, b_i)$, with the new colour value of the pixel $O_{new}(i, j)$ defined as

$$r \stackrel{def}{=} (\tfrac{r_c + r_{c'}}{2} + r_i)/2,\ g \stackrel{def}{=} (\tfrac{g_c + g_{c'}}{2} + g_i)/2,\ b \stackrel{def}{=} (\tfrac{b_c + b_{c'}}{2} + b_i)/2.$$

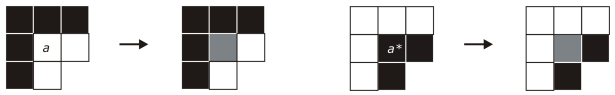


Figure 5.3: Step 1 worked out for the structuring elements (A_1, B_1).

Step 2. In step 2 we detect corners with the pair (A_2, B_2). If we detect such a foreground corner a or a background corner a^* at position (i, j) in O (see figure 5.4), we change the corner pixels at positions $(i-1, j)$ and $(i, j-1)$ or at positions $(i-1, j-1)$ and (i, j) in the blown up image O_{new}. First we determine if the pixel value $O(i-1, j)$ is equal to the pixel value $O(i, j-1)$. Analogously for $O(i-1, j-1)$ and $O(i, j)$. If $O(i-1, j) = O(i, j-1)$ and $O(i-1, j-1) \neq O(i, j)$, then we fill the pixels at positions $(i-1, j-1)$ and (i, j) with the pixel value $O(i-1, j)$. If $O(i-1, j-1) = O(i, j)$ and $O(i-1, j) \neq O(i, j-1)$, then we fill the pixels at positions $(i-1, j)$ and $(i, j-1)$ with the pixel value $O(i-1, j-1)$. If $O(i-1, j-1) = O(i, j)$ and $O(i-1, j) = O(i, j-1)$, we determine which of the two pixel values $O(i-1, j)$ and $O(i-1, j-1)$ is the foreground colour. Note that the foreground colour is defined by the minority colour in the image. So we determine which of the two values $O(i-1, j)$ and $O(i-1, j-1)$ is less present in the image. If $O(i-1, j)$ is the foreground colour, we fill the pixels at positions $(i-1, j-1)$ and (i, j) with $O(i-1, j)$. If $O(i-1, j-1)$ is the foreground colour, we fill the pixels at positions $(i-1, j)$ and $(i, j-1)$ with $O(i-1, j-1)$. If there is no minority colour $O(i-1, j)$ or $O(i-1, j-1)$ in the image, that is, if there are as many colour pixels $O(i-1, j)$ as colour pixels $O(i-1, j-1)$ in the image, we leave the colour values of the pixels unchanged.

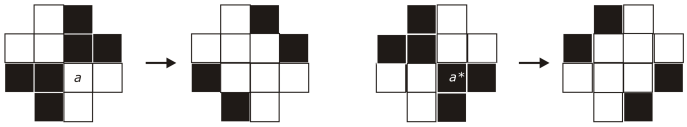


Figure 5.4: Step 2 illustrated for the structuring elements (A_2, B_2).

Step 3. We have added the structuring elements (A_3, B_3), (A_4, B_4), (A_5, B_5) and rotated or reflected structuring elements because we experienced that they are representative for the corner structures that should not be changed in an image. When we find a corner determined by one of these structuring elements, we leave the observed pixels unchanged in O_{new} to avoid that real corners will be removed in the magnified image.

Step 4. We look at structuring elements of the form (A_6, B_6), (A_7, B_7) and rotated or reflected versions, see figure 5.5. In the first case (1) when a foreground corner a or a background corner a^* is determined by (A_6, B_6) at position (i, j) in O, we replace the pixel value at position $(i + 1, j - 1)$ or at position $(i + 1, j)$ in the image O_{new} by the colour with RGB components
for $O_{new}(i + 1, j - 1)(r, g, b)$:

$$r \stackrel{def}{=} (3/4 \cdot r_c + 1/4 \cdot r_i),\ g \stackrel{def}{=} (3/4 \cdot g_c + 1/4 \cdot g_i),\ b \stackrel{def}{=} (3/4 \cdot b_c + 1/4 \cdot b_i),$$

for $O_{new}(i + 1, j)(r, g, b)$:

$$r \stackrel{def}{=} (1/4 \cdot r_c + 3/4 \cdot r_i),\ g \stackrel{def}{=} (1/4 \cdot g_c + 3/4 \cdot g_i),\ b \stackrel{def}{=} (1/4 \cdot b_c + 3/4 \cdot b_i),$$

where $[r_i, g_i, b_i]$ and $[r_c, g_c, b_c]$ are the RGB colour vectors of the pixels $O(i, j)$ and $O(i, j - 1)$. In the second case (2), if a corner a or a^* is determined by the structuring elements (A_7, B_7) or by the structuring elements (A_6, B_6) in the special composition as illustrated in figure 5.5(a) for a (for a^* we get a similar figure) at position (i, j) in O, we replace the original colour at position $(i + 1, j)$ or at position $(i + 1, j - 1)$ in O_{new} by the colour with RGB components
for $O_{new}(i + 1, j - 1)(r, g, b)$:

$$r \stackrel{def}{=} (3/4 \cdot r_c + 1/4 \cdot r_i),\ g \stackrel{def}{=} (3/4 \cdot g_c + 1/4 \cdot g_i),\ b \stackrel{def}{=} (3/4 \cdot b_c + 1/4 \cdot b_i),$$

for $O_{new}(i + 1, j)(r, g, b)$:

$$r \stackrel{def}{=} (1/4 \cdot r_c + 3/4 \cdot r_i),\ g \stackrel{def}{=} (1/4 \cdot g_c + 3/4 \cdot g_i),\ b \stackrel{def}{=} (1/4 \cdot b_c + 3/4 \cdot b_i),$$

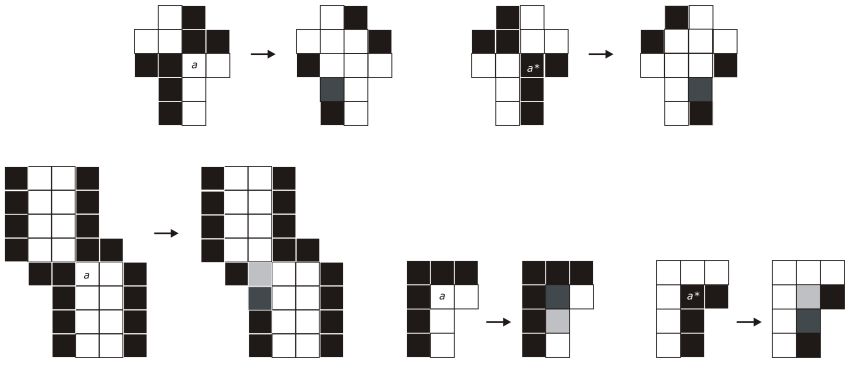
where $[r_i, g_i, b_i]$ and $[r_c, g_c, b_c]$ are the RGB colour vectors of the pixels $O(i, j)$ and $O(i, j - 1)$. The colour value of $O_{new}(i, j)$ or $O_{new}(i, j - 1)$ is changed to the RGB

colour (r', g', b') with
for $O_{new}(i, j-1)(r', g', b')$:

$$r' \stackrel{def}{=} (1/4 \cdot r_c + 3/4 \cdot r_i), g' \stackrel{def}{=} (1/4 \cdot g_c + 3/4 \cdot g_i), b' \stackrel{def}{=} (1/4 \cdot b_c + 3/4 \cdot b_i),$$

for $O_{new}(i, j)(r', g', b')$:

$$r' \stackrel{def}{=} (3/4 \cdot r_c + 1/4 \cdot r_i), g' \stackrel{def}{=} (3/4 \cdot g_c + 1/4 \cdot g_i), b' \stackrel{def}{=} (3/4 \cdot b_c + 1/4 \cdot b_i).$$



(a)

Figure 5.5: Step 4, at the top: case (1) and at the bottom: case (2).

Step 5. At last we consider the pairs of structuring elements (A_8, B_8), (A_9, B_9) and their rotated or reflected structuring elements. If we find such a foreground corner a (or a background corner a^*) at position (i, j), then we move the colour of pixel $O_{new}(i+1, j-1)$ ($O_{new}(i+1, j)$) to the pixel $O_{new}(i+2, j-1)$ ($O_{new}(i+2, j)$) and give pixel $O_{new}(i+1, j-1)$ ($O_{new}(i+1, j)$) an intermediate colour value between the colours $O_{new}(i, j-1)$ and $O_{new}(i+2, j-1)$ or $O_{new}(i, j)$ and $O_{new}(i+2, j)$, that is, $O_{new}(i+1, j-1) = O_{new}(i, j-1) + O_{new}(i+2, j-1)$ and $O_{new}(i+1, j) = O_{new}(i, j) + O_{new}(i+2, j)$, as shown in figure 5.6.

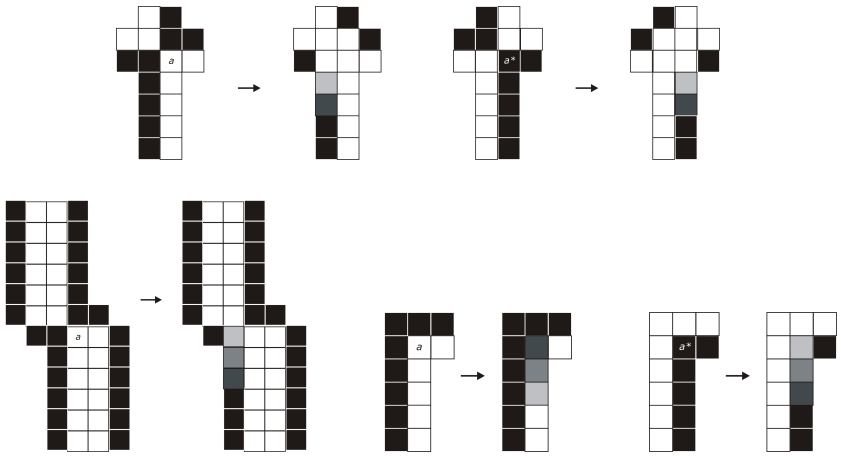


Figure 5.6: Step 5, at the top: case (1) and at the bottom: case (2).

Colour images

When the original image X is a colour image, the blown up image O and the complement $co(O)$ of O will also be colour images. The inner edge-image O_{edge} of O and the inner edge-image O^c_{edge} of $co(O)$ are originally both colour images, but we transform them into greyscale images. The definition of the hit-or-miss operator as intersection of two erosions can be extended to greyscale images, using the threshold approach for example. The hit-or-miss operator based on the threshold approach of a greyscale image I by two binary structuring elements A and B is defined as

$$I \otimes_t (A, B) = E_t(I, A) \cap E_t(co(I), B),$$

with $co(I)$ the complementary image of I. In our method we determine the inner edge-image (the intern morphological gradient) of a colour image X and the inner edge-image of the complementary image $co(X)$ of X to detect corners in the image. It is important and even necessary that the used colour vector ordering $\leq_{RGB}$ is compatible with the complement co to get 'supplementary' inner edge-images for X and $co(X)$.

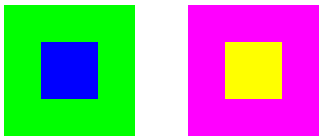


Figure 5.7: The original image X (left) and the complementary image $co(X)$ of X (right).

In figure 5.7 you see the original image X and its complement $co(X)$. When we use an ordering that is not compatible with the complement co, we can get results as in figure 5.8 (upper line) for the inner edge-image of X and $co(X)$, where it is impossible to detect corners using the hit-or-miss transform. Figure 5.8 (lower line) shows the t-'colour' erosions of X and $co(X)$ and their corresponding induced intern morphological gradients.

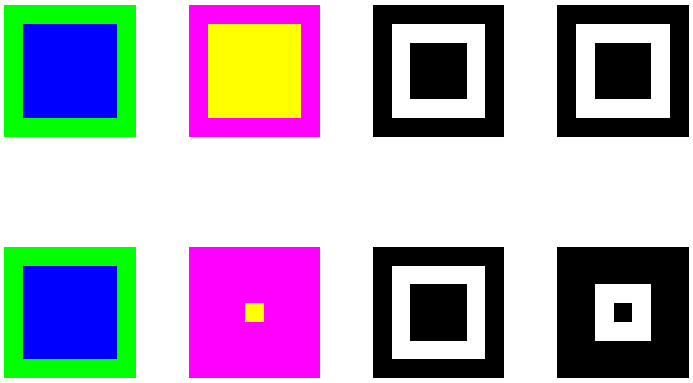


Figure 5.8: From left to right: above: possible t-erosions of X and $co(X)$ by B^{Wh}_{RGB} and the corresponding inner edge images, when using an ordering that is not compatible with the complement; below: the t-'colour' erosions $\vec{E}_t(X, B^{Wh}_{RGB})$ and $\vec{E}_t(co(X), B^{Wh}_{RGB})$ and their corresponding intern morphological gradients $\vec{G}^{B^{Wh}_{RGB}}_{t,i}(X)$ and $\vec{G}^{B^{Wh}_{RGB}}_{t,i}(co(X))$, using $\leq_{RGB}$.

For the corner correction we can work here in the same way as described before for binary images. The hit-or-miss transformations $O_{edge} \otimes_t (A_i, B_i)$ and $O^c_{edge} \otimes_t (A_i, B_i)$, for $i = 1 \ldots 9$, are now greyscale images so that we get for every pixel in O a value between 0 and 1 that gives us a degree of being a corner edge pixel for that pixel. The property 'being a corner' can thus be evaluated on a numeric scale, where the fuzzily known values for this property can be represented as a fuzzy set on $\mathbb{R}$. Therefore we have constructed an S-membership function, by experiment, as you can see in figure 5.9, with parameters $\alpha = 0.15$, $\gamma = 0.3$ and $\beta = \frac{\alpha+\gamma}{2}$, given by

$$\begin{array}{rccll} S(.;\alpha,\beta,\gamma): & \mathbb{R} & \rightarrow & [0,1] & \\ & x & \mapsto & 0, & \forall x \in [-\infty, \alpha] \\ & x & \mapsto & 2(\frac{x-\alpha}{\gamma-\alpha})^2, & \forall x \in [\alpha, \beta] \\ & x & \mapsto & 1 - 2(\frac{x-\gamma}{\gamma-\alpha})^2, & \forall x \in [\beta, \gamma] \\ & x & \mapsto & 1, & \forall x \in [\gamma, +\infty]. \end{array}$$

The fuzzy property 'being a corner' is represented by means of the membership function $S(.; 0.15, 0.225, 0.3)$. Every number below $\alpha = 0.15$ does not satisfy the property 'being a corner' at all and every number beyond the value $\gamma (> \alpha) = 0.3$ satisfies the property 'being a corner' completely. Our first idea was to work with weights in

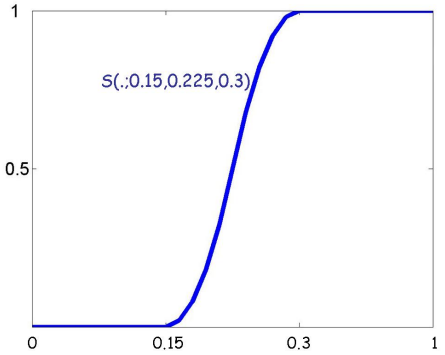


Figure 5.9: Graphical representation of the S-membership function $S(.; 0.15, 0.225, 0.3)$.

the corner correction procedure to obtain good results. We look back at step 1 in our corner correction method, where we determine corners using the structuring elements (A_1, B_1) and their rotations. Instead of changing the colour at position (i, j) in O by $(O(i-1, j)(r_c, g_c, b_c) + O(i, j-1)(r_{c'}, g_{c'}, b_{c'})) + O(i, j)(r_i, g_i, b_i)$ we now add a weight w, determined by $O_{edge} \otimes_t (A_1, B_1)(i, j)$ or $O^c_{edge} \otimes_t (A_1, B_1)(i, j)$, and change the pixel values, here explained for foreground corner pixels determined by $O_{edge} \otimes_t (A_1, B_1)(i, j)$, but it is analogous for background corner pixels determined by $O^c_{edge} \otimes_t (A_1, B_1)(i, j)$, as follows

$O = double(O)$;
$[mO, nO, 3] =$ size(O);
$O_{new} = O$;
for $s = 1 : mO$
  for $ss = 1 : nO$
    $w(s, ss) = S(O_{edge} \otimes_t (A_1, B_1)(s, ss)(r); 0.15, 0.225, 0.3)$
    $= S(O_{edge} \otimes_t (A_1, B_1)(s, ss)(g); 0.15, 0.225, 0.3)$
    $= S(O_{edge} \otimes_t (A_1, B_1)(s, ss)(b); 0.15, 0.225, 0.3)$
    if $w(s, ss) >= 0.3$
      for $sss = 1 : 3$
        $O_{new}(s, ss)(sss) = ((O(s-1, ss) + O(s, ss-1)) + O(s, ss))(sss)$;
      end;
    else if $w(s, ss) < 0.3$ & $w(s, ss) >= 0.15$
        for $sss = 1 : 3$
          $O_{new}(s, ss)(sss) = (w(s, ss) \cdot (O(s-1, ss) + O(s, ss-1))$
            $+ (1 - w(s, ss)) \cdot O(s, ss))(sss)$;
        end;
    else
        for $sss = 1 : 3$
          $O_{new}(s, ss)(sss) = O(s, ss)(sss)$;
        end;
      end;
    end;
  end;
end;

In the other steps of the corner correction we can work in the same way.

However, experiments have shown that this does not necessarily give better results, on the contrary, as when we do not take the weight w into account. And this is because of the following. Consider a pixel at position (i, j) in O. When we add the weight $w(i, j)$ to the mixture $(O(i-1, j)(r_c, g_c, b_c)+O(i, j-1)(r_{c'}, g_{c'}, b_{c'}))+O(i, j)(r_i, g_i, b_i)$, i.e., $w(i, j)\cdot(O(i-1, j)+O(i, j-1))+(1-w(i, j))\cdot O(i, j)$, the smaller the value for $w(i, j)$ (the larger the value for $1 - w(i, j)$), the less difference in colour there will be between the mixture $w(i, j) \cdot (O(i-1, j) + O(i, j-1)) + (1 - w(i, j)) \cdot O(i, j)$ and $O(i, j)$. Consequently, this mixture will become less 'strong' in the sense that it is possible that no or almost no difference in colour is visible anymore between this mixture and $O(i, j)$, and thus changing the pixel value at position (i, j) has had no effect, although the corner in (i, j) has to be smoothed since it causes a staircasing effect. So we better do not take the weight w into account in the corner transformation. The smaller the value in (i, j) for the property 'being a corner', the less clear a corner is visible at position (i, j) in O and the less difference in colour is noticeable between $O(i, j)$ and the surrounding pixel values; the larger the value in (i, j) for the property 'being a

corner', the more clear a corner is visible at position (i,j) in O and the more difference in colour is noticeable between $O(i,j)$ and the surrounding pixel values. When we change the pixel values in (i,j) by $(O(i-1,j)(r_c,g_c,b_c)+O(i,j-1)(r_{c'},g_{c'},b_{c'}))+O(i,j)(r_i,g_i,b_i)$, we will get a gradual corner correction transition from pixels not being a corner to pixels being a corner.

We have made the choice, by experiments, that if $w(i,j) \leq 0.15$, then there is no corner or the corner-property is very weak at position (i,j) in O so that we do not chance the pixel values in (i,j), because otherwise too much pixels in the image would be changed creating a blurring effect. If $w(i,j) > 0.15$, then there is a visible corner at position (i,j) in O, which has to be smoothed, because otherwise jagged edges would remain in the image. We get:

$O = double(O)$;
$[mO, nO, 3] =$ size(O);
$O_{new} = O$;
for $s = 1 : mO$
  for $ss = 1 : nO$
    $w(s,ss) = S(O_{edge} \otimes_t (A_1, B_1)(s,ss)(r); 0.15, 0.225, 0.3)$
    $= S(O_{edge} \otimes_t (A_1, B_1)(s,ss)(g); 0.15, 0.225, 0.3)$
    $= S(O_{edge} \otimes_t (A_1, B_1)(s,ss)(b); 0.15, 0.225, 0.3)$
    if $w(s,ss) >= 0.15$
      for $sss = 1 : 3$
        $O_{new}(s,ss)(sss) = ((O(s-1,ss) + O(s,ss-1)) + O(s,ss))(sss)$;
      end;
    else
      for $sss = 1 : 3$
        $O_{new}(s,ss)(sss) = O(s,ss)(sss)$;
      end;
    end;
  end;
end;

In the other steps of the corner transformation we work in the same way.

5.2.4 Magnification by an Integer Factor $n > 2$

Now, for magnification by an integer factor $n > 2$, we have to extend the structuring elements to a larger size but a similar shape, and the way of filling up the edge pixels will change a bit, but is analogous. In figure 5.10 we have illustrated this process for magnification by a factor 3. In figure 5.11 and 5.12 you see our corner correction for

magnification by a factor 4 and 5, where we have replaced the colour values in a similar way. And so you can go on for magnification by a larger integer factor.

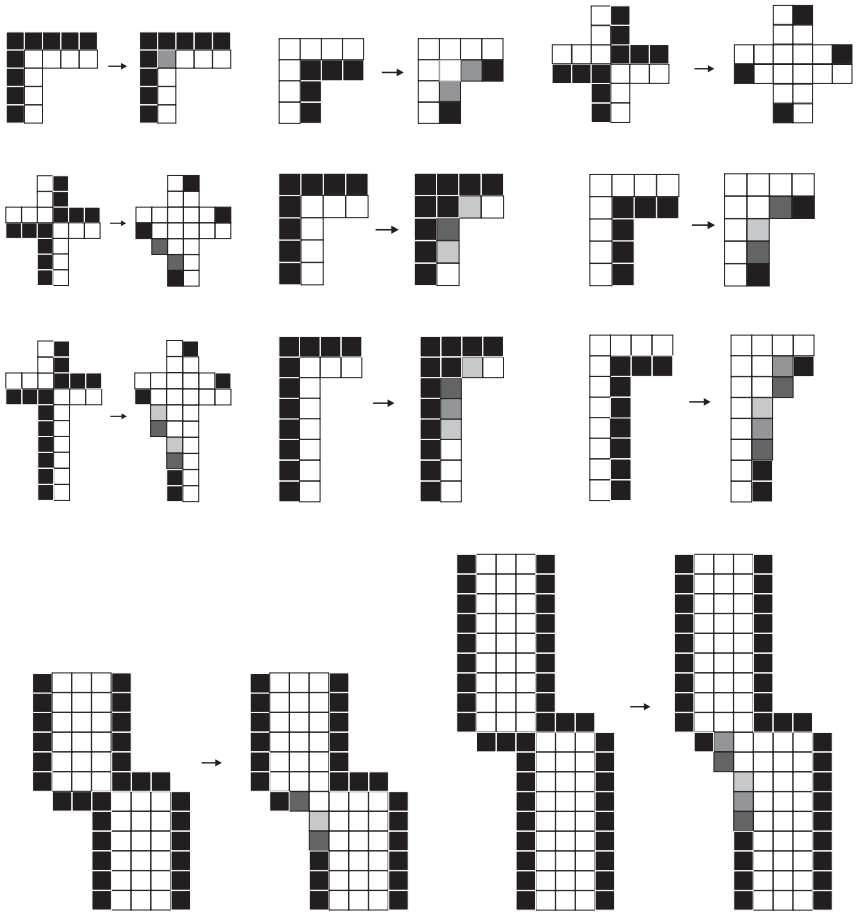


Figure 5.10: Corner correction for magnification by a factor 3.

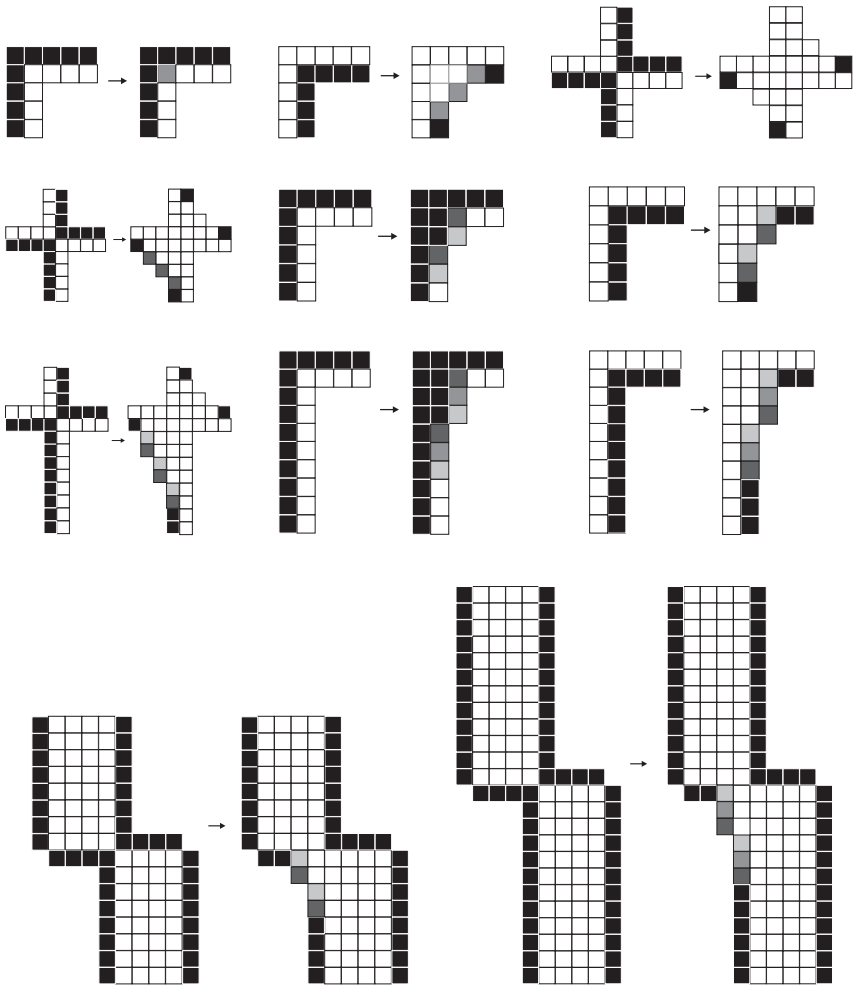


Figure 5.11: Corner correction for magnification by a factor 4.

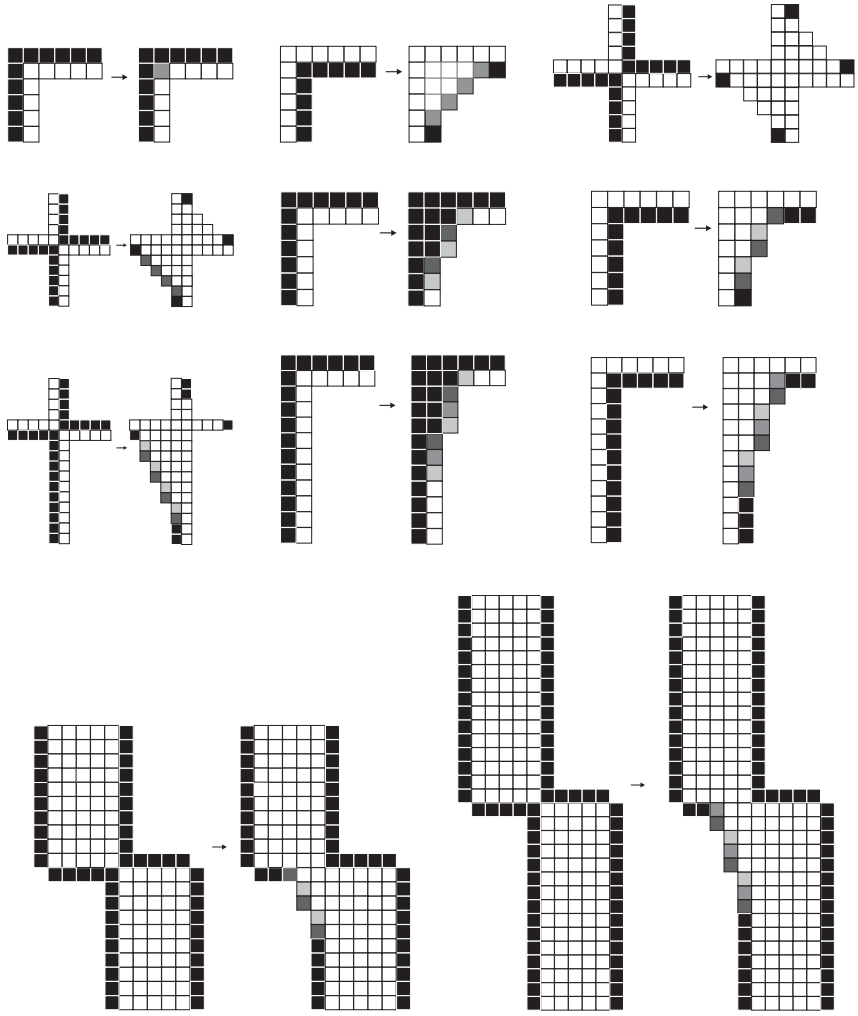


Figure 5.12: Corner correction for magnification by a factor 5.

5.2.5 Experimental Results

Figures 5.13 to 5.16 show some results of our interpolation method.

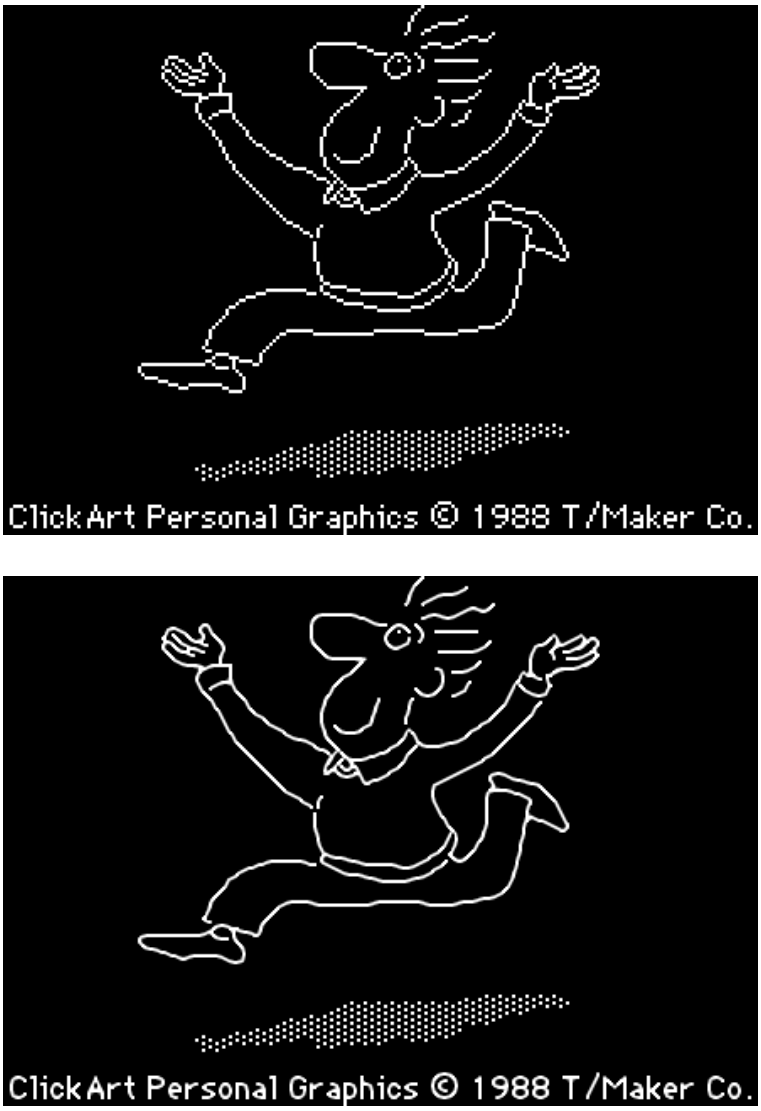


Figure 5.13: At the top: the pixel replicated 'cartoon' image for magnification by a factor 2, at the bottom: the result of our morphological interpolation method.

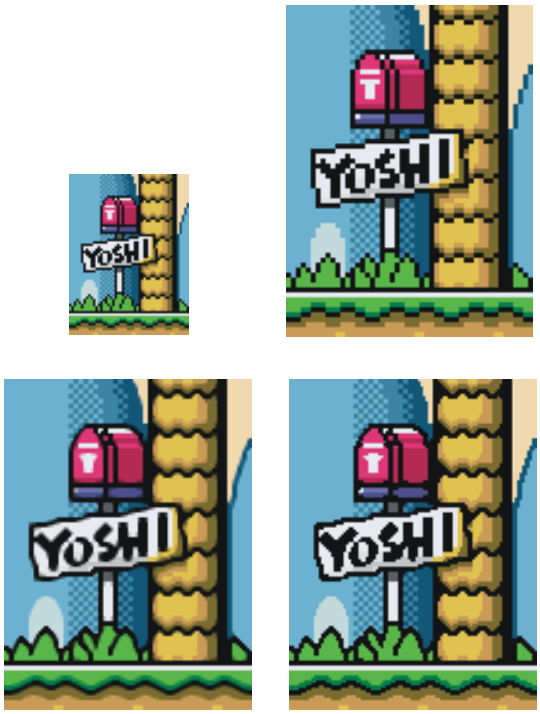


Figure 5.14: At the top: the original 'mailbox' image (left) and the nearest neighbour interpolated image for magnification by a factor 2 (right), at the bottom: the result of our new morphological interpolation method (left) and our method, giving as result a colour image with no new colours (right).

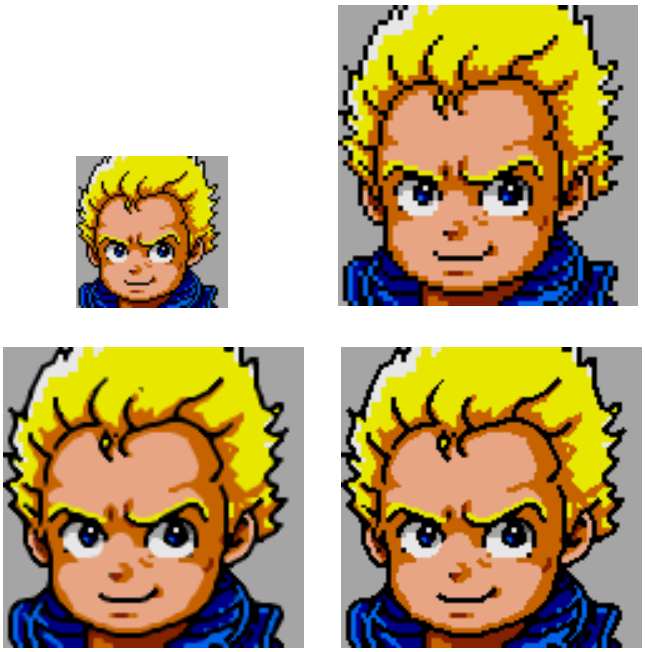


Figure 5.15: At the top: the original ‘man’ image (left) and the nearest neighbour interpolated image for magnification by a factor 2 (right), at the bottom: our new morphological interpolation method (left) and the result of our method introducing no new colours (right).

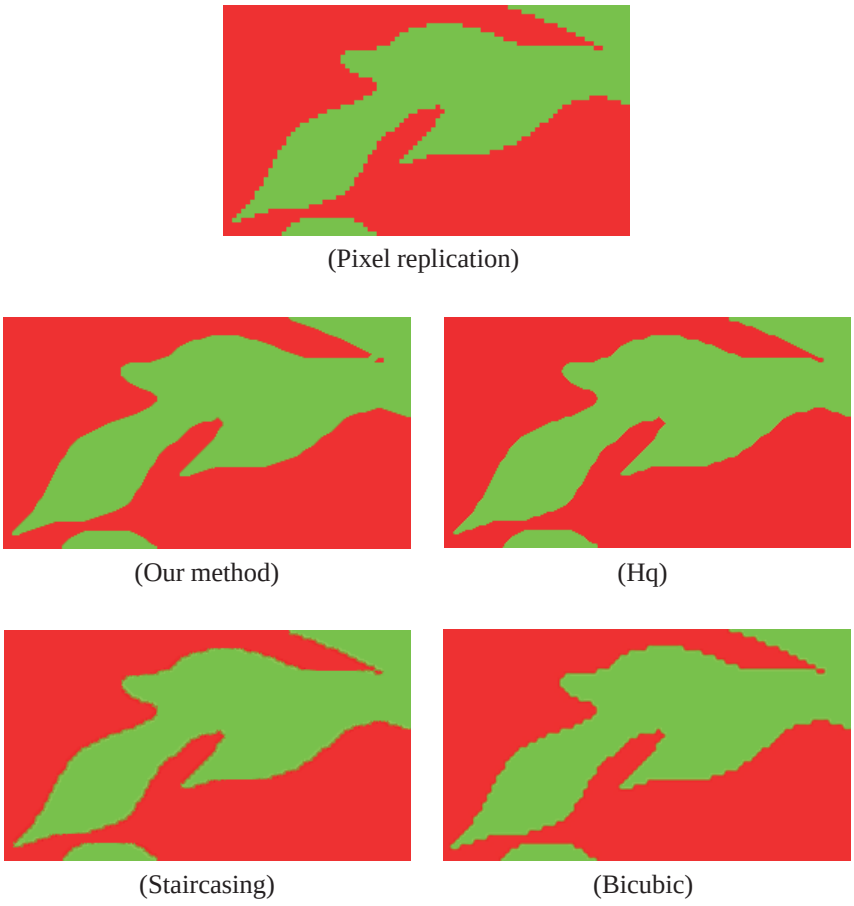


Figure 5.16: Interpolation results for magnification by a factor 3.

Figure 5.16 and 5.17 illustrate the result of several interpolation methods. We have compared our technique with the following state-of-the-art methods: the high-quality magnification filter Hq [76], which analyses a 3×3 area around the source pixel and makes use of lookup tables to get interpolated pixels of the filtered image, the staircasing filter [54], which detects staircasing and changes only staircased edges and pixels, and some classical linear interpolation methods [32], in particular, bilinear and bicubic interpolation, which use the (weighted) mean of respectively 4 and 16 closest neighbours to calculate the new pixel values, and sinc interpolation [32], which makes use of windowed sinc functions. The main advantages and drawbacks of these linear interpolation filters are pointed out in [32].

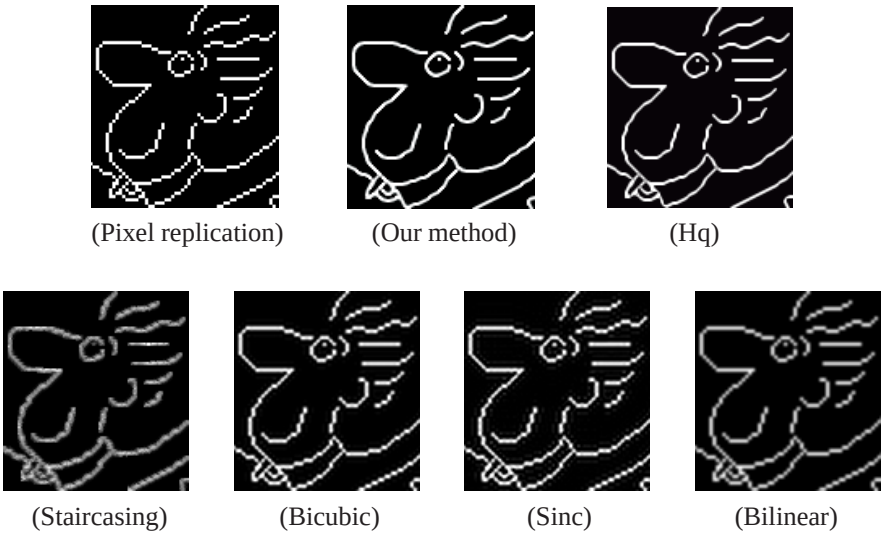


Figure 5.17: Result of several interpolation methods for magnification by a factor 2.

We may conclude that our new method provides very good results. Improvements in visual quality can be noticed: unwanted jaggies have been removed so that edges have become smoother and the edges are very sharp and clear. Good results are also obtained with the Hq interpolation method, but our method outperforms all the others.

Our method was implemented in Matlab, which makes it hard to compare the computational complexity of this method with the others. As future work we can reimplement all methods in the same program language, Java or C++, to make them comparable with each other.

Remark: Sometimes it is desired that binary logos, cartoons and maps remain binary or that no new colours are introduced in a colour image after magnification. Our method can also produce such a result, we only have to insert a threshold in the corner correction method. In the first step all new coloured pixels with a value greater than or equal to $(O(i-1,j)+O(i,j-1))+O(i,j)$ are assigned to the background colour of that pixel, while all other pixel values remain unchanged. In the fourth and fifth step all pixels with RGB colour values greater than or equal to $(1/4 \cdot O(i,j-1)+3/4 \cdot O(i,j))$ are transformed to $O(i,j)$, the values of the colour pixels greater than or equal to $(3/4 \cdot O(i,j-1)+1/4 \cdot O(i,j))$ are transformed to $O(i,j-1)$, for $i = 6,7,8,9$, and the pixel values greater than or equal to ($[r,g,b]$ of the foreground colour pixel $+$ $[r,g,b]$ of the background colour pixel) are assigned to the background colour value of that pixel. The main visual improvements can be seen in figure 5.14, 5.15 and 5.18: the contours are smooth and text is also still interpolated very well.

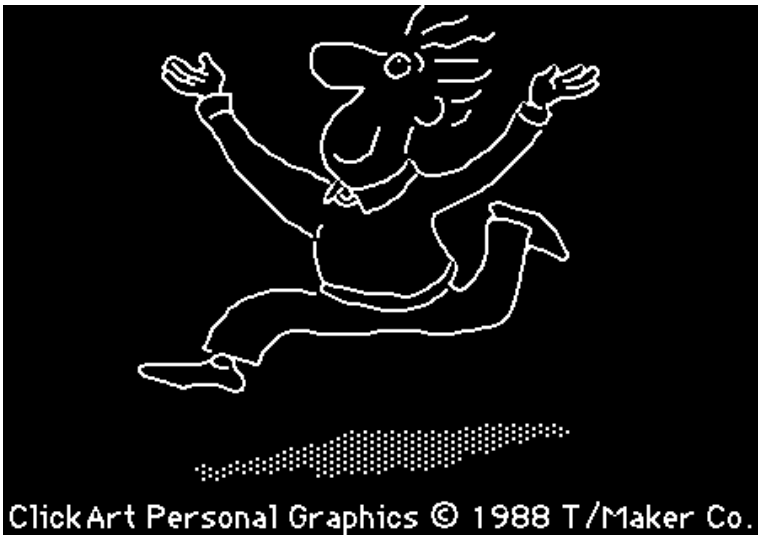


Figure 5.18: Our new morphological interpolation method, giving as result a binary image of the original binary image 'cartoon' (magnification by a factor 2).

For a binary image interpolation method making use of mathematical morphology and giving a black-and-white result, we also refer to [31]. An extension of this approach to greyscale images can be found in [30].

5.2.6 Conclusion

We have presented a magnification method that improves the visual quality of magnified binary images with sharp boundaries. The problem of interpolating an image is the introduction of unwanted jagged edges in the blown up image. We developed a new approach to avoid these jaggies, making use of mathematical morphology and our ordering $\leq_{RGB}$. We have demonstrated that our method gives beautiful results for the magnification of colour images with sharp edges. Our next step is to extend our approach towards all colour images with 'vague' edges, again using mathematical morphology.

5.3 New Morphological Image Interpolation Method to Magnify Images with Vague Edges

5.3.1 Corner Detection

Again, we use the pixel replicated or nearest neighbour interpolated image O of an original image X as first 'trivial' interpolation step. Next we determine the inner edge-image of the blown up image O, using structuring element B^{Wh}_{RGB}, in order to obtain the positions of the corner pixels of unwanted jagged edges. Let us call O_{edge} the inner edge-image of O, transformed into a greyscale image, as described in section 5.2.2. Since we work here with 'vague' edges, it is much more difficult to detect the unwanted jaggies. In contrast with sharp edges, vague edges are usually less fine, less clear and less sharp, whereby pixels along vague edges will have a lower degree of being a corner edge pixel, which makes it more difficult to detect them. First we have tried to apply the hit-or-miss operator to O and $co(O)$ to detect edge points of jaggies, but we did not succeed in detecting all unwanted corner pixels. The condition that a pixel has to belong to the intersection of two erosions is too strict to detect all vague edges. So let us take a closer look at the inner edge-image O_{edge} of a blown up image O.

In figure 5.19 you see an example of a nearest neighbour interpolated image O and the internal morphological gradient O_{edge} of O by B^{Wh}_{RGB}. When we look at O_{edge}, we see a kind of cross structure appearing at the edge where the pixels $O_{edge}(3,7)$, $O_{edge}(5,9)$ and $O_{edge}(7,11)$ indicate unwanted jaggies. Accordingly, we got the idea to work with one type of binary structuring elements, see figure 5.20, where A contains the white pixels $(\times)$ and B the white pixels $(\bullet)$, to detect these edge pixels.

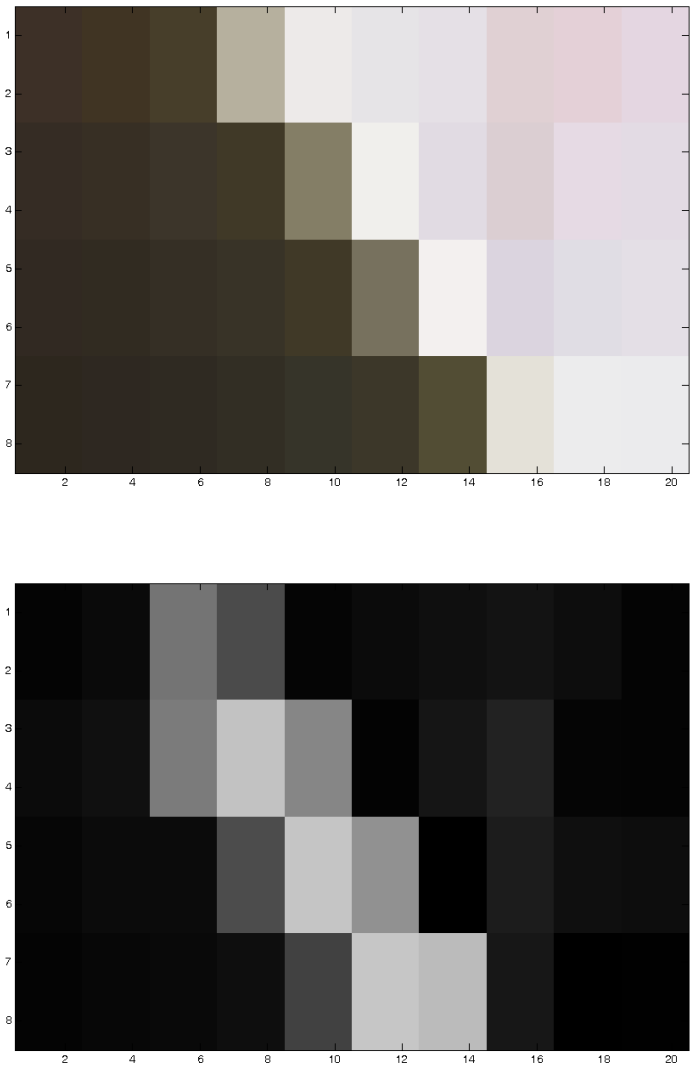


Figure 5.19: The pixel replicated image O (at the top) and the inner edge-image O_{edge} of O by B^{Wh}_{RGB}(at the bottom).

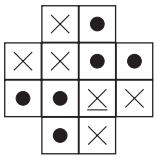
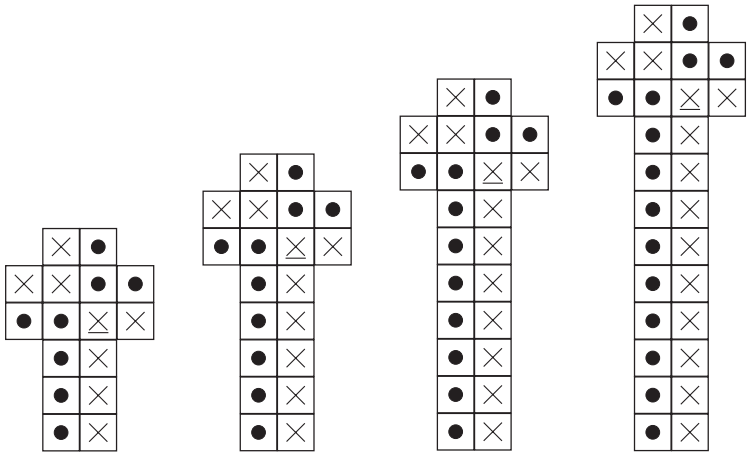


Figure 5.20: The used binary structuring elements (A, B) (the underlined element corresponds to the origin of coordinates) for magnification by a factor 2.

With this given pair of structuring elements (A, B) we determine the erosions $E_t(O_{edge}, A)$ and $E_t(O_{edge}, B)$ and then take in every pixel position (i, j) of O the maximum of $E_t(O_{edge}, A)(i, j)$ and $E_t(O_{edge}, B)(i, j)$. The erosion $E_t(O_{edge}, A)$ allows us to detect jaggies in the $\backslash$-slanted direction, call this the second diagonal direction, while the erosion $E_t(O_{edge}, B)$ allows us to detect jaggies in the $/$-slanted direction, call this the first diagonal direction. The structuring elements we have used further on are illustrated in figure 5.21. The others are rotated or reflected versions of these.



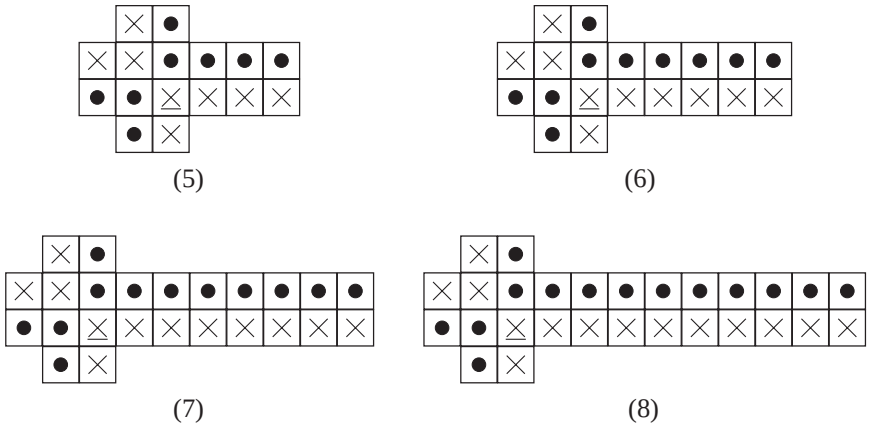


Figure 5.21: The used binary structuring elements (1) (A_1, B_1) ... (8) (A_8, B_8) (the underlined element corresponds to the origin of coordinates) for magnification by a factor 2, where A_i contains the white pixels ($\times$) and B_i the white pixels ($\bullet$), $i = 1 \ldots 8$.

5.3.2 Corner Correction

In this section we describe our corner correction method, which is again a trade-off between blur and jaggies, for magnification by a factor 2.

When we detect a corner at position (i, j) in O, we look in which direction, along the first or second diagonal, the edge moves and determine how strong the jaggie is by calculating the weight $w(i, j)$. As explained in section 5.2.3 for colour images, when $w(i, j) \leq 0.15$, we will leave the pixel values around $O(i, j)$ unchanged, and when $w(i, j) > 0.15$, we will change the pixel values around $O(i, j)$. Depending on in which direction the edge moves, we will change the pixel values in and around $O(i, j)$ in a different way. When the edge moves in the second diagonal slanted direction, we look for that side of the edge of which the colour lies closest for the human eye to the colour value of the pixel $O(i, j)$. We do this by transforming the colour image O modelled in RGB into the L*a*b* colour model, call this image $O_{L^*a^*b^*}$, and then calculating

$$\begin{aligned} north = \max(&norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i-v,j)),\\ &norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j+v)));\\ south = \max(&norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i+v,j)),\\ &norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j-v)));\end{aligned}$$

where v is the integer factor by which we have magnified the image. If $north > south$, then we will change pixels at the right side of the pixel $O(i, j)$, and if $north < south$,

then we will change pixels at the left side of the pixel $O(i,j)$. Analogously, when the edge moves in the first diagonal slanted direction, we look for the side of the edge of which the colour lies closest for the human eye to the colour value of the pixel $O(i,j)$ by calculating

$$\begin{aligned} north = \max(&norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i-v,j)),\\ &norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j-v)));\\ south = \max(&norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i+v,j)),\\ &norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j+v))). \end{aligned}$$

If $north > south$, then we will change pixels at the left side of the pixel $O(i,j)$, and if $north < south$, then we will change pixels at the right side of the pixel $O(i,j)$.

If $north = south$, in the second diagonal slanted direction or in the first diagonal slanted direction, we calculate

$$\begin{aligned} north1 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i-v,j)),\\ south2 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j-v)),\\ north2 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j+v)),\\ south1 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i+v,j)). \end{aligned}$$

SECOND DIAGONAL SLANTED DIRECTION

$(north > south)$ or
$((north = south)$ and $((north1 > south1)$ or $(north2 > south2)))$

Step 1. When we detect a corner at position (i,j) in O, denoted by $\leftthreetimes$, along the second diagonal and where $north > south$ or $((north = south)$ and $((north1 > south1)$ or $(north2 > south2)))$, we first look in which direction, to the right or going down, the difference in colour between $O(i,j)$ and the neighbouring pixels is largest, by calculating

$$\begin{aligned} S_1 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i,j+v))\\ S_2 &= norm(O_{L^*a^*b^*}(i,j) - O_{L^*a^*b^*}(i+v,j)). \end{aligned}$$

Dependent on in which direction we get the largest colour difference, we will change other pixel values, as shown in figure 5.22.

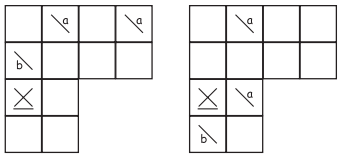


Figure 5.22: Corner correction to the right if $S_1 > S_2$ (left) or going down if $S_1 < S_2$ (right).

If $S_1 > S_2$, we look at the pixels $O(i-1,j), O(i-2,j+1)$ and $O(i-2,j+3)$. If $S_1 < S_2$, we look at the pixels $O(i+1,j), O(i,j+1)$ and $O(i-2,j+1)$. The colour of the pixels at position $(i-1,j)$ or $(i+1,j)$ in O is replaced by a mixture of the adjoining pixels, that is, we add the pixel values at positions (i,j) and $(i-1,j-1)$ or at positions $(i+2,j)$ and $(i+1,j-1)$. Let $[r_c, g_c, b_c]$ and $[r_{c'}, g_{c'}, b_{c'}]$ be the RGB colour vectors of the pixels $O(i,j)$ and $O(i-1,j-1)$ or $O(i+2,j)$ and $O(i+1,j-1)$ respectively. If we call our new image O_{new}, then $O_{new}(i-1,j)(r,g,b) = (O(i,j)(r_c,g_c,b_c) + O(i-1,j-1)(r_{c'},g_{c'},b_{c'}))$ or $O_{new}(i+1,j)(r,g,b) = (O(i+2,j)(r_c,g_c,b_c) + O(i+1,j-1)(r_{c'},g_{c'},b_{c'}))$, with the new colour values $[r,g,b]$ defined as

$$r \stackrel{def}{=} \frac{r_c+r_{c'}}{2},\ g \stackrel{def}{=} \frac{g_c+g_{c'}}{2},\ b \stackrel{def}{=} \frac{b_c+b_{c'}}{2}.$$

The colour of the pixels at positions $(i-2,j+1)$ and $(i-2,j+3)$ or $(i,j+1)$ and $(i-2,j+1)$ in O_{new} is also replaced by a mixture of the adjoining pixels. For example, for the pixels $O(i-2,j+1)$ and $O(i-2,j+3)$ we get $O_{new}(i-2,j+1) = (O(i-3,j+1)+O(i-2,j+2))$ and $O_{new}(i-2,j+3) = (O(i-3,j+3)+O(i-2,j+4))$. We have used different notations to denote which pixels should be mixed: when one of the adjoining pixels is lying above the considered pixel, and the other adjoining pixel is in this case then lying at the right side, we denote ◹, and when one of the adjoining pixels is lying below the considered pixel, and the other adjoining pixel here now at the left side, we denote ◺. If $S_1 = S_2$, we change the pixels as in the cases $S_1 > S_2$ and $S_1 < S_2$.

Thereafter we detect corners with the structuring element A_5 if $S_1 > S_2$ or with the structuring element A_1 if $S_1 < S_2$. If $S_1 = S_2$, we detect corners with the structuring elements A_1 and A_5 and look for which structuring element we get the largest value.

Step 2. In step 2 we calculate $E_t(O_{edge}, A_5)(i,j)$ and verify if $E_t(O_{edge}, A_5)(i,j) > \alpha$ and $\neg(E_t(O_{edge}, A)(i+2,j+2) > E_t(O_{edge}, A)(i,j+2))$. When both conditions are fulfilled, we may assume that there is a step at position (i,j) in the staircasing edge that moves further to the right, as at position $(9,5)$ in O_{edge} shown in figure 5.23. Thus we will smooth the jagged edge at (i,j) further in the right direction, as shown in figure 5.24(a).

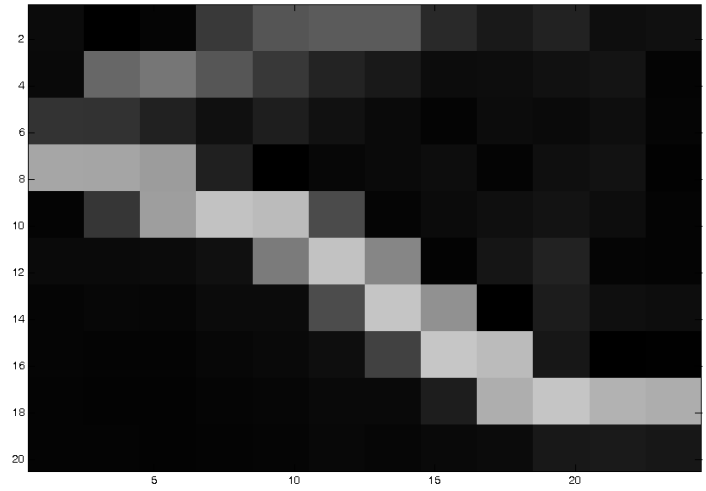


Figure 5.23: An example of an inner edge-image O_{edge} where we see an edge point at position $(9, 5)$ moving to the right.

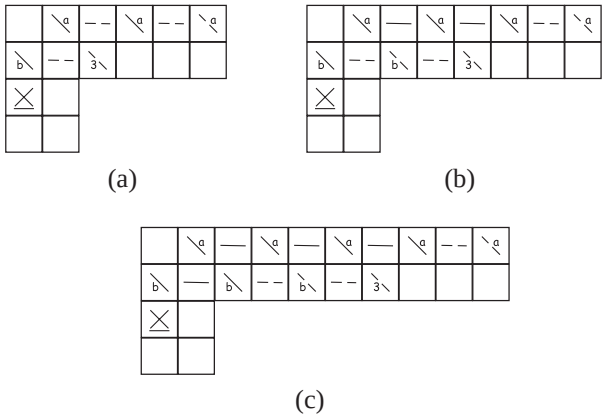


Figure 5.24: Corner transformation in step 2 for the structuring elements A_5, A_6 and A_7.

First we change the pixel value at position $(i-1, j+2)$ in O_{new} by a mixture, denoted by $\diagdown\!\!\!\diagdown$, of the colour $O(i-1, j+2)$ and the two adjoining colours $O(i-1, j+1)$ and $O(i, j+2)$, i.e., $O_{new}(i-1, j+2)(r, g, b) = (O(i-1, j+2)(r_c, g_c, b_c) + O(i-1, j+1)(r_{c'}, g_{c'}, b_{c'}) + O(i, j+2)(r_{c''}, g_{c''}, b_{c''}))$, where

$$r \stackrel{def}{=} \frac{r_c + r_{c'} + r_{c''}}{3},\ g \stackrel{def}{=} \frac{g_c + g_{c'} + g_{c''}}{3},\ b \stackrel{def}{=} \frac{b_c + b_{c'} + b_{c''}}{3}.$$

Thereafter we replace the pixel value $O_{new}(i-1, j+1)$ by an intermediate colour value between the colours $O_{new}(i-1, j)$ and $O_{new}(i-1, j+2)$, i.e., $O_{new}(i-1, j+1) = O_{new}(i-1, j) + O_{new}(i-1, j+2)$, which we indicate by $-$. Next the pixel value $O_{new}(i-2, j+5)$ is replaced by a mixture of the adjoining pixels $O(i-2, j+6)$ and $O(i-3, j+5)$, again denoted by $\diagdown\!\!\!\diagdown$. And at last the pixel values $O_{new}(i-2, j+2)$ and $O_{new}(i-2, j+4)$ are changed to an intermediate colour between the new colour values $O_{new}(i-2, j+1)$ and $O_{new}(i-2, j+3)$ for $O_{new}(i-2, j+2)$ and $O_{new}(i-2, j+3)$ and $O_{new}(i-2, j+5)$ for $O_{new}(i-2, j+4)$, again denoted by $-$.

Analogously we calculate $E_t(O_{edge}, A_6)(i, j)$ and verify if $E_t(O_{edge}, A_6)(i, j) > \alpha$ and $\sim (E_t(O_{edge}, A)(i+2, j+4) > E_t(O_{edge}, A)(i, j+4))$. When both conditions are fulfilled, we know that at position (i, j) there is a step in the staircasing effect that moves still further on to the right so that we have to smooth the jagged edge further to the right, and so we can go on, for example for the structuring element A_7 as illustrated in figure 5.24(c).

Step 3. For steps moving downstairs in the staircasing edge, determined by the structuring elements A_1, A_2, A_3 and A_4, we work completely analogously as in step 2. The corner correction here is illustrated in figure 5.25.

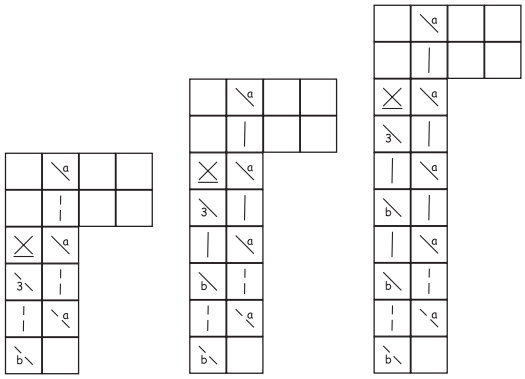
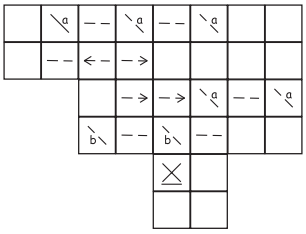
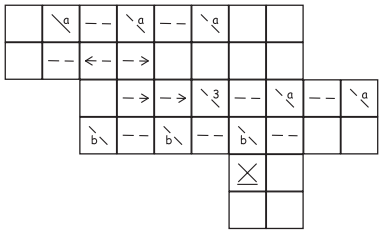


Figure 5.25: The corner correction in step 3 worked out for the structuring elements A_1, A_2 and A_3.

Step 4. Finally in the last two steps 4 and 5 we check if the jagged corners follow each other. When we detect a jagged edge point at position (i, j), we look if there is an edge point present at position $(i-2, j-2)$. If so, then the jagged corners follow each other. If not, it is possible that there is a jagged edge point present at position $(i-2, j-4)$ or $(i-4, j-2)$ or at position $(i-2, j-6)$ or $(i-6, j-2)$. If we find an edge point at position $(i-2, j-4)$ that does not move to the right and is or is not coming from an edge moving to the right, or if we find an edge point at position $(i-2, j-6)$ that moves one time to the right and is or is not coming from an edge moving to the right, then we change the pixels around $O_{new}(i, j)$ as illustrated in figure 5.26(a). If there is no edge point present at position $(i-2, j-4)$ and we find an edge point at position $(i-2, j-6)$ that does not move to the right or an edge point at position $(i-2, j-8)$ that is moving one time to the right, then we change the pixels around $O_{new}(i, j)$ as illustrated in figure 5.26(b). The symbol ←- denotes that we add the colour of that pixel to the colour of the left pixel, while the symbol -→ denotes that we add the colour of that pixel to the colour of the right pixel. The other symbols have been explained in the previous steps.



(a)



(b)

Figure 5.26: Step 4, at the top: case (a) and at the bottom: case (b).

Step 5. In figure 5.27 we illustrate how we replace the pixel values when there is no edge point present at position $(i-2, j-2)$ and we detect a jagged edge point at position $(i-4, j-2)$ that does not move downstairs or at position $(i-6, j-2)$ that moves one time downstairs (see figure 5.27(a)), or when there is no edge point present at positions

$(i-2, j-2)$ and $(i-4, j-2)$ and we find an edge point at position $(i-6, j-2)$ that does not move downstairs or at position $(i-8, j-2)$ that moves one time downstairs (see figure 5.27(b)).

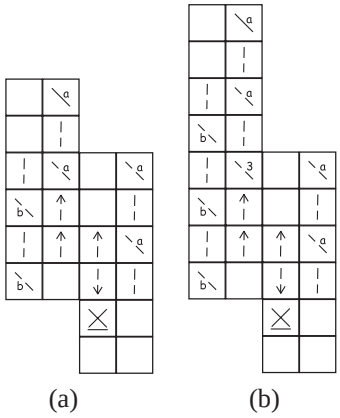
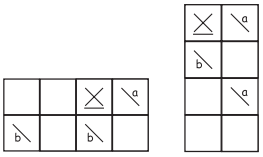


Figure 5.27: Step 5, at the left: case (a) and at the right: case (b).

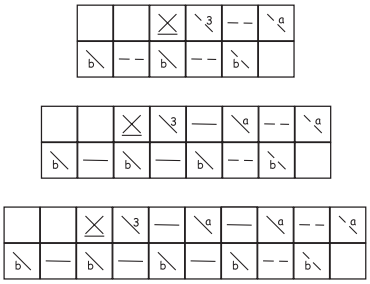
In the other cases, the second diagonal slanted direction with $(north < south)$ or $((north = south)$ and $((north1 < south1)$ or $(north2 < south2)))$ and the first diagonal slanted direction with $(north > south)$ or $((north = south)$ and $((north1 > south1)$ or $(north2 < south2)))$ and $(north < south)$ or $((north = south)$ and $((north1 < south1)$ or $(north2 > south2)))$, we work completely analogously as explained before so that we will only give the figures that illustrate which pixels are changed and how they are replaced.

$(north < south)$ or
$((north = south)$ and $((north1 < south1)$ or $(north2 < south2)))$

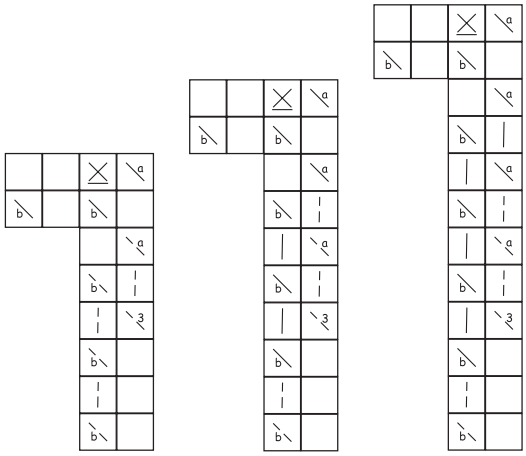
Step 1.



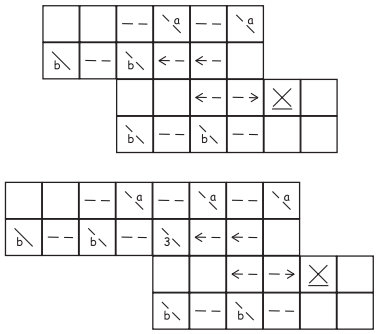
Step 2.



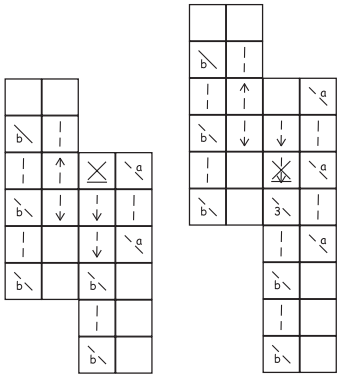
Step 3.



Step 4.



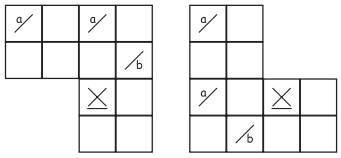
Step 5.



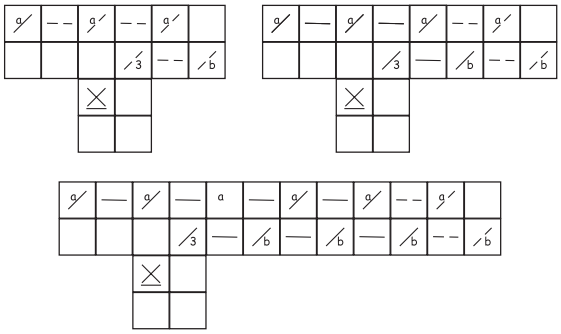
FIRST DIAGONAL SLANTED DIRECTION

$(north > south)$ **or**
$((north = south)$ **and** $((north1 > south1)$ **or** $(north2 < south2)))$

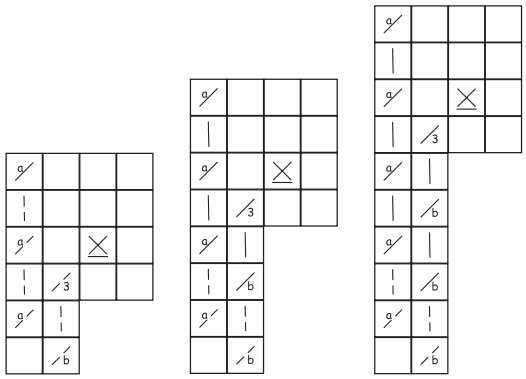
Step 1.



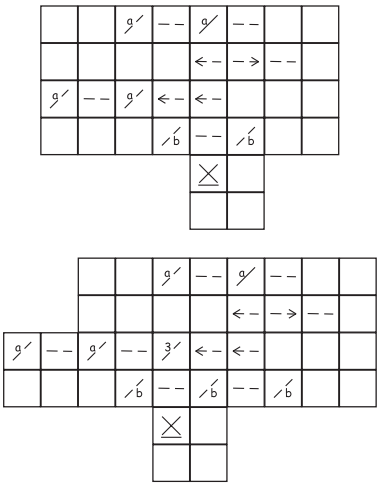
Step 2.



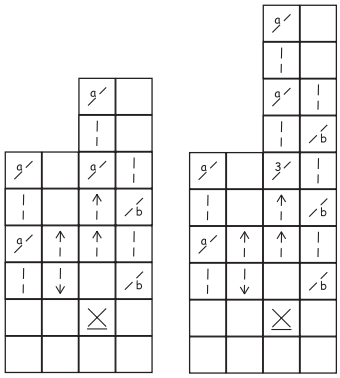
Step 3.



Step 4.

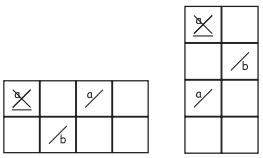


Step 5.

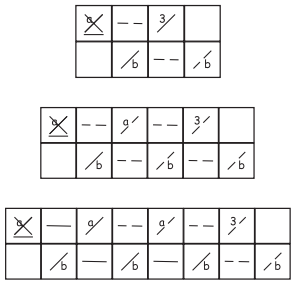


$(north < south)$ **or**
$((north = south)$ **and** $((north1 < south1)$ **or** $(north2 > south2)))$

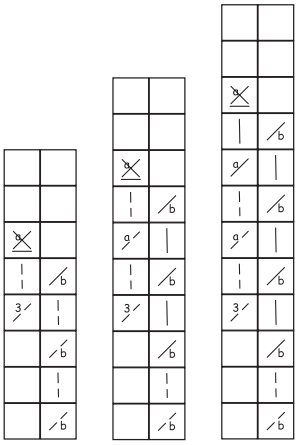
Step 1.



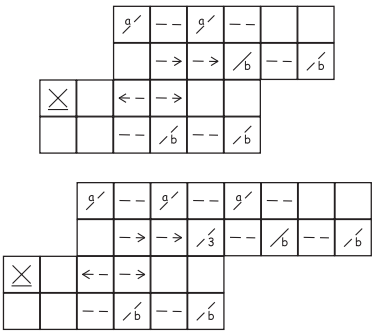
Step 2.



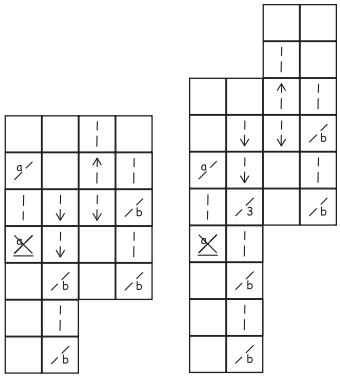
Step 3.



Step 4.



Step 5.



Lastly, when all cases have been considered in our method, we combine our result with the bilinear interpolation method, as follows. Let us call O_{bil} the bilinear interpolated image of X for magnification by a factor 2. For every position (i, j) in O we look if pixels have been changed in the 5×5 neighbourhood of $O(i, j)$ with our method. If no pixels have been changed in this neighbourhood, then we replace the pixel values of O_{new} in these surroundings by the corresponding pixel values in O_{bil}. We do this because our interpolation method works very well along edges between areas of objects in the image with a different colour. But our method will not be able to detect corners in regions with little detail, because these edges are too weak, although they also have to be smoothed to avoid jaggies when we magnify the image by a large factor. Combination of both our morphological interpolation method and the bilinear interpolation method deals with all edges.

5.3.3 Magnification by an Integer Factor $n > 2$

Now, for magnification by an integer factor $n > 2$, we have to extend the structuring elements to a larger size but a similar shape, and the way of filling up the edge pixels will change a bit, but is very similar. In figure 5.28 and 5.29 we have illustrated our corner correction process for magnification by a factor 3 and 4. And so you can go on for magnification by a larger integer factor.

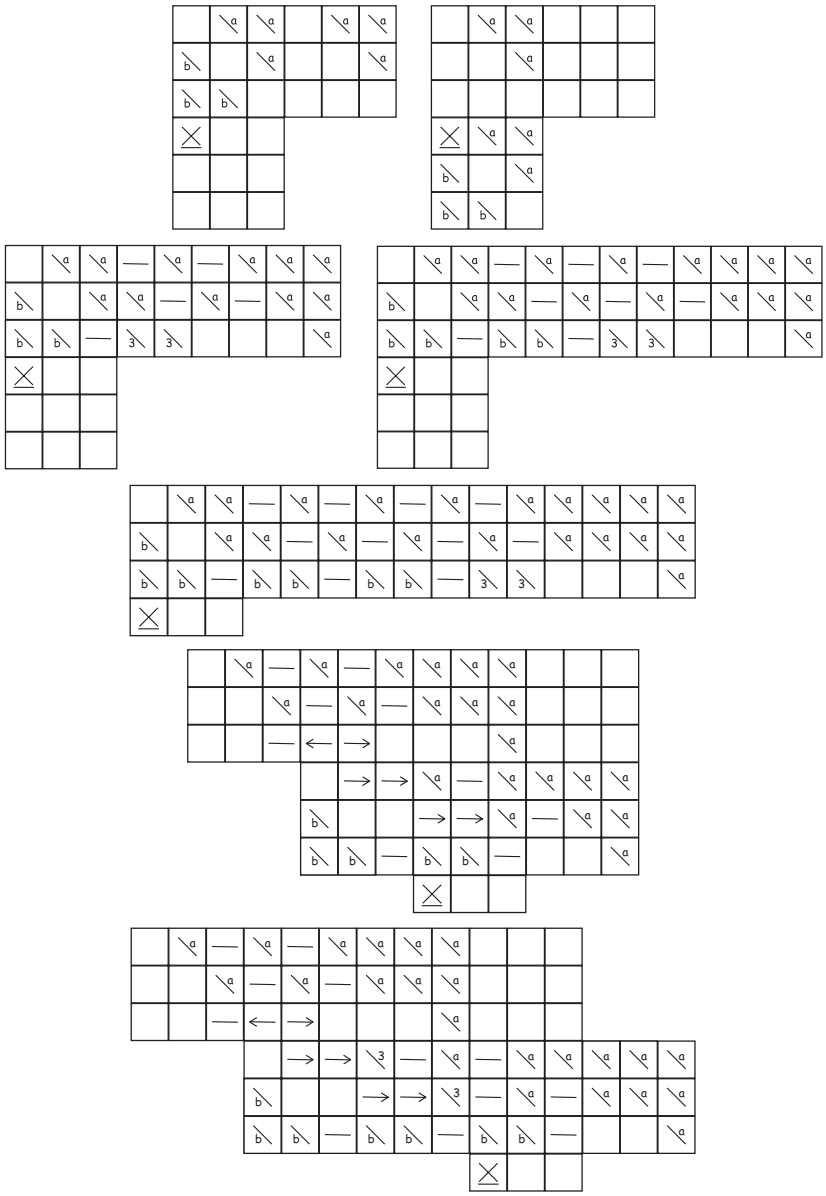


Figure 5.28: Corner correction for magnification by a factor 3.

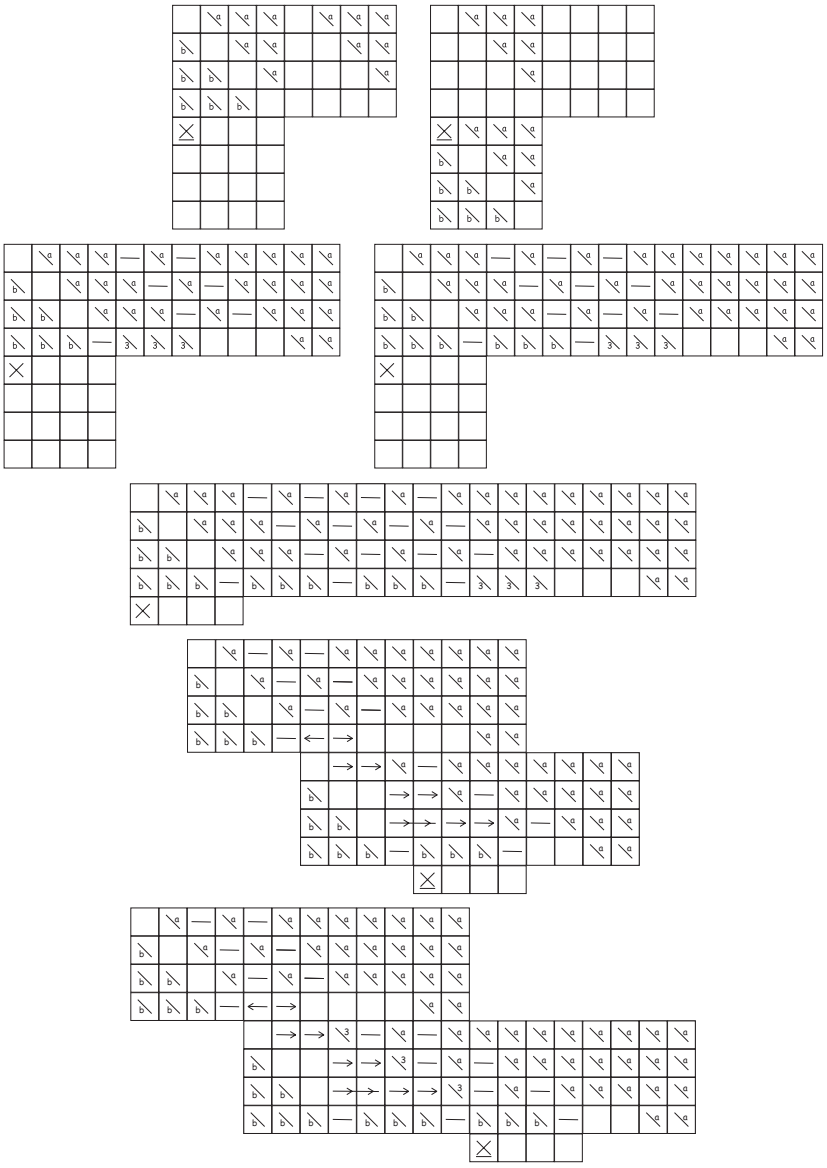


Figure 5.29: Corner correction for magnification by a factor 4.

5.3.4 Experimental Results

Figures 5.30 and 5.31 show some results for magnification by a factor 2.

Figure 5.30: Some results of our new morphological magnification method: the original colour image (left), the nearest neighbour interpolated image (middle) and our result (right).

More in detail:

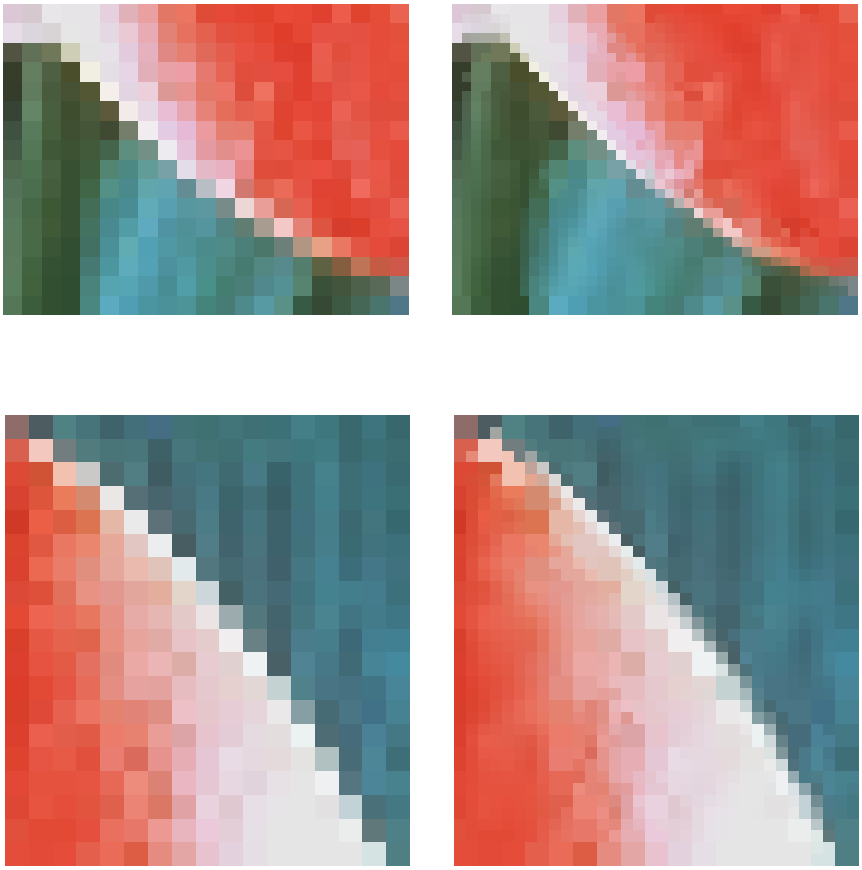


Figure 5.31: Some of the results of figure 5.30 in more detail: left: the nearest neighbour interpolated image and right: our morphological magnification result.

Figures 5.32, 5.33, 5.34 and 5.35 illustrate the result of several interpolation methods for magnification by a factor 2, 3 and 4.

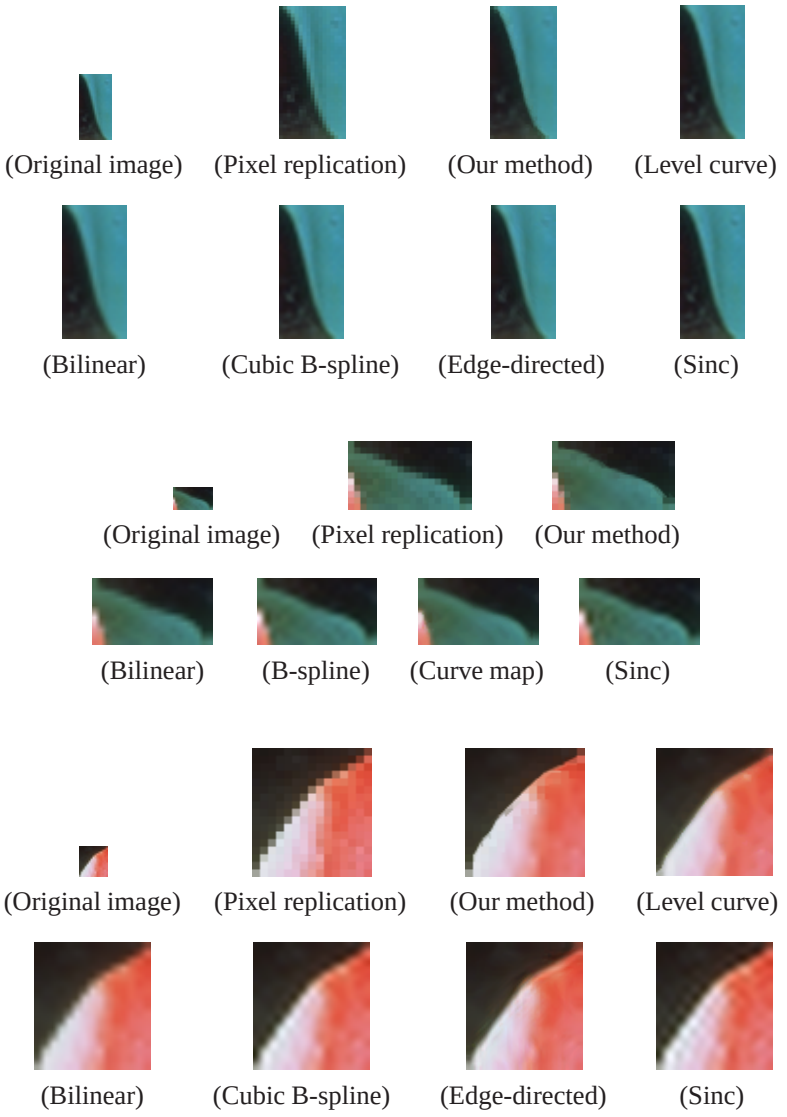
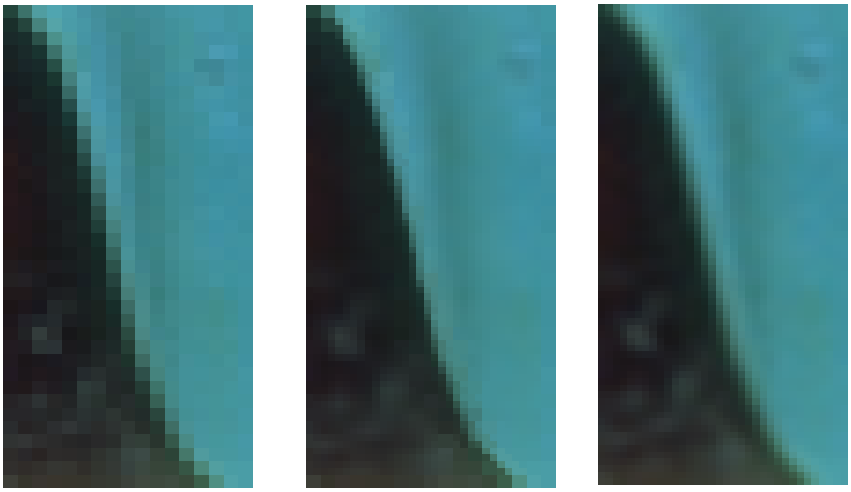
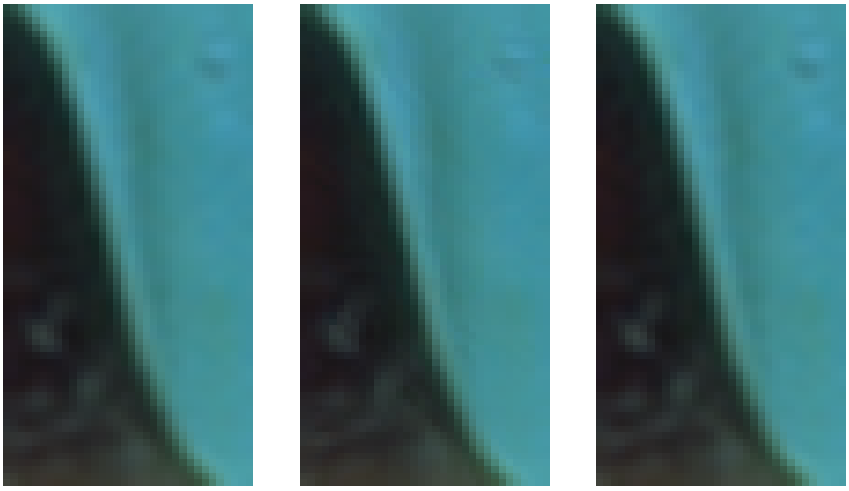


Figure 5.32: From top to bottom: result of several interpolation methods for magnification by a factor 2, 3 and 4.



(Pixel replication) (Our method) (Level curve mapping)



(Cubic B-spline) (Edge-directed) (Sinc)

Figure 5.33: Interpolation results for magnification by a factor 2, in detail.

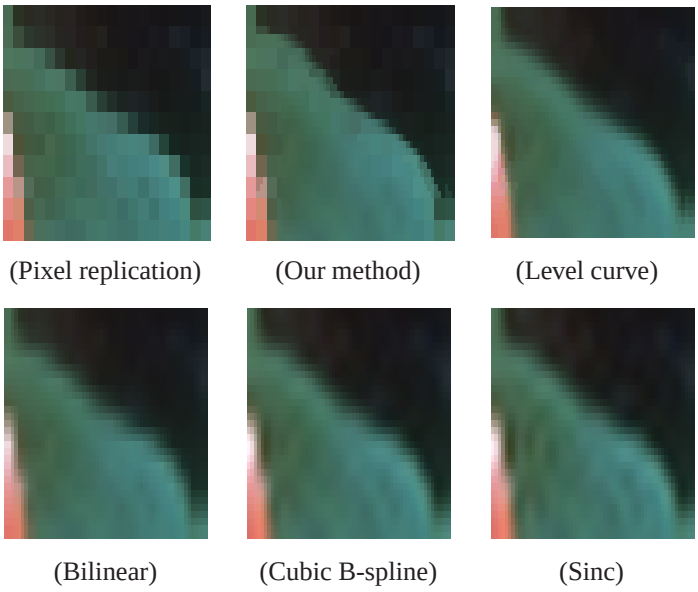


Figure 5.34: Interpolation results for magnification by a factor 3, in detail.

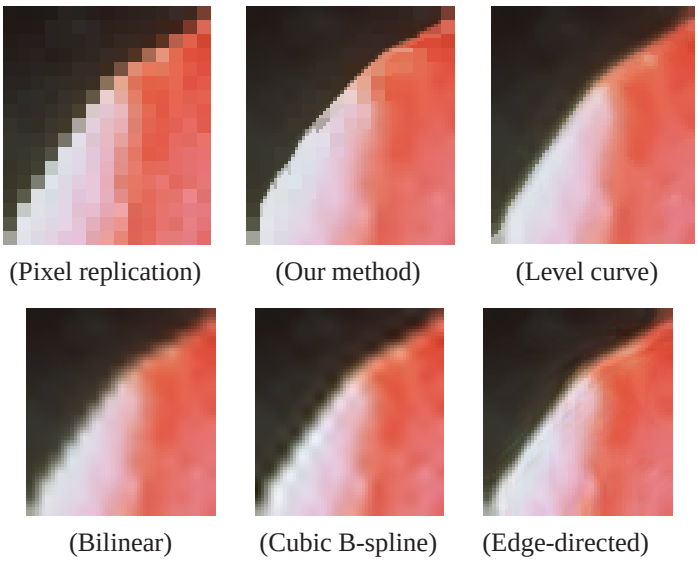


Figure 5.35: Interpolation results for magnification by a factor 4, in detail.

We have compared our technique with the following state-of-the-art methods: bilinear interpolation, which uses the (weighted) mean of 4 closest neighbours to calculate the new pixel values, cubic b-spline interpolation [80, 81], which makes use of continuous B-spline basis functions, level curve mapping [38], which sharpens edges by mapping the image level curves using adaptive contrast enhancement techniques, sinc interpolation [32], which makes use of windowed sinc functions and new edge-directed interpolation [34].

We may conclude that our new method provides very good results. Improvements in visual quality can be noticed: unwanted jaggies have been removed so that edges have become smoother and there are no ringing and blurring artefacts present in the images. The results clearly show that our technique gives the sharpest edges.

Our method was implemented in Matlab, which makes it hard to compare the computational complexity of this method with the others. As future work we can reimplement all methods in the same program language, Java or C++, to make them comparable with each other.

5.3.5 Conclusion

We have presented a magnification method using mathematical morphology that improves the visual quality of magnified colour images with 'vague' edges. The problem of interpolating an image is the introduction of unwanted jagged edges in the blown up image. We have developed a new approach to avoid these jaggies, making use of mathematical morphology and our ordering $\leq_{RGB}$, which is a trade-off between blur and jaggies.

Chapter 6

Conclusion

Mathematical morphology was originally only developed for binary images. There exist different approaches for the extension of binary morphology to morphology for greyscale images. These extensions towards colour images are not straightforward, and this is what this work is concerned with. The problem of looking for a vector ordering for colour or multivariate morphological image processing is not new and is being developed since the early 90's.

First of all we have defined a complement co and addition +, subtraction - and multiplication * operations between colours in the RGB, HSV and L*a*b* colour model. A new colour ordering in RGB, HSV and L*a*b*, which is compatible with the defined complement co, is proposed, where we have treated the colours as vectors so that we get complete lattices for these colour models seen with the new corresponding ordering. Definition of associated minimum and maximum operators follows, and so we can extend the greyscale morphological operators based on the threshold, umbra and fuzzy approach to operators acting on colour images. At last we have examined which properties, e.g. duality principle and monotonicity, still hold for the new colour morphological operators.

We may conclude that our new method provides better results than those obtained by the component-based approach and similar or better results than those obtained by other state-of-the-art methods. One great advantage is that the existing correlations between the different colour components are taken into account. And what is more, more details from the original colour image are preserved and thus visible.

As application we have used our new approach to colour morphology to magnify images. Image magnification has many applications such as simple spatial magnification of images (e.g. printing low-resolution documents on high-resolution printer devices, digital zoom in digital cameras), geometrical transformation (e.g. rotation), etc. Differ-

ent image magnification methods have already been proposed in the literature. Because the existing methods usually suffer from one or more artefacts such as staircasing and blurring, we have developed a new image interpolation method, based on mathematical morphology, to magnify binary as well as colour images with sharp edges. Finally, an extension of our morphological interpolation method to magnify colour images with vague edges is studied.

We may conclude that our new method provides very good results and an improvement w.r.t. the state-of-the-art. Improvements in visual quality can be noticed: unwanted jaggies have been removed so that edges have become smoother and there are no ringing and blurring artefacts present in the images. Our approach gives the sharpest and clearest edges.

As future work we can set up an experiment regarding the psycho visual behaviour of similarity measures, which can be useful for the evaluation of morphological operators. Our morphological image magnification method for sharp and vague edges have been implemented in Matlab, which makes it hard to compare the computational complexity of this method with the others. As future work we can reimplement all methods in the same program language, Java or C++, to make them comparable with each other.

Chapter 7

Samenvatting

Mathematische morfologie is een theorie voor de analyse van ruimtelijke structuren, gebaseerd op verzamelingenleer en het begrip verschuiving. In de jaren zestig voerden G. Matheron en J. Serra [41], beiden geïnspireerd door de studie naar de geometrische vorm van poreus medium, het begrip mathematische morfologie in. Poreus medium is binair in de zin dat een punt van poreus medium ofwel deel uitmaakt van een porie ofwel behoort tot de grondmassa rond de poriën. Zo ontwikkelden Matheron en Serra een theorie voor de analyse van binaire beelden. De grondmassa kan beschouwd worden als de verzameling van objectpunten in het beeld, terwijl de poriën het complement van deze verzameling vormen. Bijgevolg kunnen objectpunten behandeld worden met eenvoudige bewerkingen zoals unie, doorsnede, complement en verschuiving. Mathematische morfologie werd oorspronkelijk dus enkel voor binaire beelden ontwikkeld. Op deze manier legden Matheron en Serra alvast de basis voor mathematische morfologie in de beeldanalyse. Vandaag de dag heeft mathematische morfologie vele toepassingen in de beeldanalyse zoals randdetectie, ruisverwijdering, objectherkenning, patroonherkenning, beeldsegmentatie en beeldvergroting in o.a. de biologische en medische wereld [71, 74]. De basiswerktuigen van mathematische morfologie zijn de morfologische operatoren die een gegeven beeld A dat we willen analyseren omzet naar een nieuw beeld $P(A, B)$ gebruik makend van een structuurelement B, om zo bijkomende informatie over de vorm, grootte, oriëntatie of beeldafmetingen van voorwerpen in A te verkrijgen. Behalve de schijfjes- en umbrabenadering kan binaire morfologie uitgebreid worden naar morfologie voor grijswaardenbeelden door gebruik te maken van vaagverzamelingenleer, vaagmorfologie genoemd. De toepassing van morfologische operatoren op kleurenbeelden is zeker niet voor de hand liggend. En daarover handelt dit proefschrift.

In hoofdstuk 2 beginnen we met de voorstelling van digitale beelden en enkele definities in verband met vaagverzamelingen, vaaglogische operatoren, $\mathcal{L}$-vaagverzamelingen, $\mathcal{L}$-vaaglogische operatoren, $\mathcal{L}$-vaagrelaties en $\mathcal{L}$-vaagrelationele beelden die we

verder in dit proefschrift nodig hebben. In het derde hoofdstuk leggen we uit hoe het menselijk oog is opgebouwd en hoe het licht, en zo ook kleur, waarneemt. Daarna beschrijven we de additieve en subtractieve kleurenmengeling om kleur te reproduceren en leggen we het verschil uit tussen de termen kleurenmodel en kleurenruimte. In hoofdstuk 3 bestuderen we ook de kleurenmodellen RGB, CMY en CMYK, YUV, YIQ en YCbCr, HSV en HSL, CIEXYZ, CIEYxy, L*a*b* en L*u*v*. De definitie van de fundamentele morfologische operatoren dilatatie en erosie wordt ingevoerd in hoofdstuk 4. We beschouwen zowel de binaire morfologische operatoren als de grijswaardenmorfologische operatoren gebaseerd op de schijfjes- en umbrabenadering en de vaagverzamelingenleer. Vervolgens geven we een overzicht en een korte bespreking van de bestaande uitbreidingen van mathematische morfologie naar kleur uit de literatuur. Een eerste manier om de morfologische operatoren voor grijswaardenbeelden toe te passen op kleurenbeelden is de componentsgewijze aanpak, waarbij we de morfologische operatoren op elk van de kleurencomponenten afzonderlijk bewerken. Maar deze aanpak leidt vaak tot artefacten in het beeld omdat geen rekening gehouden wordt met de samenhang tussen de kleurencomponenten. Daarom hebben we gezocht naar een vectorordening voor kleuren, waarbij we de RGB, HSV en L*a*b* kleurenmodellen hebben beschouwd. Nadien hebben we minimum en maximum operatoren gedefinieerd en nieuwe $+$, $-$ en $*$ bewerkingen tussen kleuren zodat we de grijswaardenmorfologische operatoren konden uitbreiden naar nieuwe vectorgebaseerde operatoren voor kleurenbeelden. Het probleem van het zoeken naar een vectorordening voor kleur is niet nieuw en werd reeds in het begin van de jaren negentig ontwikkeld. Wat nieuw is hier is de gebruikte aanpak, namelijk door de umbramethode en de vaagverzamelingenleer. Experimentele resultaten tonen aan dat we zeer goede resultaten bekomen. Tenslotte hebben we in hoofdstuk 5 onze nieuwe kleurenmorfologische aanpak toegepast op het vergroten van beelden. Verschillende methodes voor het vergroten van beelden werden reeds voorgesteld in de literatuur [2, 6, 20, 27, 32, 34, 38, 42, 54, 76, 78], maar we hebben echter slechts één artikel [1] gevonden dat gebruik maakt van mathematische morfologie. Omdat de bestaande methoden meestal tekortschieten door hoek- en verwazigingsartefacten, hebben we een nieuwe methode ontwikkeld voor het vergroten van zwart-wit beelden en kleurenbeelden met scherpe randen, gebaseerd op mathematische morfologie. Daar een eenvoudige vergroting van een beeld kartelingen introduceert, zal onze methode deze gekartelde randen opsporen en afronden. Experimenten illustreren dat onze methode zeer goed presteert voor de interpolatie van scherpe beelden zoals logo's, cartoons en kaarten, voor zwart-wit beelden en kleurenbeelden. Tot slot hebben we onze morfologische interpolatiemethode uitgebreid voor het vergroten van kleurenbeelden met onscherpe randen. Hierbij bekomen we mooie resultaten die een verbetering zijn van de bestaande methoden.

Tenslotte merken we op dat bepaalde onderdelen van dit proefschrift reeds gepubliceerd werden in een boek [15] en in internationale tijdschriften [14, 16] en een aantal onderzoeksresultaten gepresenteerd werden op (internationale) conferenties [13, 14,

16]. Bijdragen tot het werk van anderen werden gepubliceerd in een boek [69], internationale tijdschriften [30, 46, 48, 49, 60, 61, 62, 64, 65, 66, 68] en proceedings van (internationale) conferenties [29, 31, 45, 47, 50, 63, 67, 82, 83, 84, 86, 85].

Bibliography

[1] ALBIOL, A., AND SERRA, J. Morphological image enlargements. *Journal of Visual Communication and Image Representation 8*, 4 (1997), 367–383.

[2] ALLEBACH, J., AND WONG, P. Edge-directed interpolation. In *Proceedings of IEEE International Conference on Image Processing ICIP 1996* (Switzerland, September 1996), pp. 707–710.

[3] ANGULO, J. Unified morphological color processing framework in a lum/sat/hue representation. In *Proceedings of International Symposium on Mathematical Morphology ISMM 2005* (France, April 2005), pp. 387–396.

[4] ANGULO, J. Morphological colour operators in totally ordered lattices based on distances: application to image filtering, enhancement and analysis. *Computer Vision and Image Understanding doi:10.1016/j.cviu.2006.11.08* (2007).

[5] ANGULO, J., AND SERRA, J. Morphological coding of color images by vector connected filters. In *Proceedings of IEEE 7th International Symposium on Signal Processing and Its Applications ISSPA 2003* (France, July 2003), pp. 69–72.

[6] CAMBRIDGE UNIVERSITY. *Digital image interpolation*. http://www.cambridgeincolour.com/tutorials/image-interpolation.htm.

[7] COLLEGE OF SAINT BENEDICT - SAINT JOHN'S UNIVERSITY. *Color depth and color spaces*. http://www.csbsju.edu/itservices/teaching/c_space/colors.htm.

[8] COMER, M. L., AND DELP, E. J. Morphological operations for color image processing. *Journal of Electronic Imaging 8*, 3 (1999), 279–289.

[9] DE BAETS, B. *Uncertainty Analysis in Engineering and Sciences: Fuzzy Logic, Statistics, and Neural Network Approach*. Kluwer Academic Publishers, 1997, ch. Fuzzy morphology: a logical approach, pp. 53–67.

[10] DE BAETS, B., KERRE, E. E., AND GUPTA, M. The fundamentals of fuzzy mathematical morphology part 1: Basic concepts. *International Journal of General Systems 23* (1995), 155–171.

[11] DE BAETS, B., KERRE, E. E., AND GUPTA, M. The fundamentals of fuzzy mathematical morphology part 2: Idempotence, convexity and decomposition. *International Journal of General Systems 23* (1995), 307–322.

[12] DE COCK, M. *Een grondige studie van linguïstische wijzigers in de vaagverzamelingenleer*. PhD Thesis, Ghent University, Belgium, 2002.

[13] DE WITTE, V. Colour morphology: shortly after take-off. In *Nederlands-Belgisch Mathematisch Congres* (The Netherlands, 2004).

[14] DE WITTE, V., SCHULTE, S., KERRE, E. E., LEDDA, A., AND PHILIPS, W. Morphological image interpolation to magnify images with sharp edges. *Lecture Notes in Computer Science 4141* (2006), 381–393.

[15] DE WITTE, V., SCHULTE, S., NACHTEGAEL, M., MÉLANGE, T., AND KERRE, E. E. *Computational Intelligence Based on Lattice Theory*. Springer-Verlag, in press, ch. A lattice-based approach to mathematical morphology for greyscale and colour images.

[16] DE WITTE, V., SCHULTE, S., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. Vector morphological operators for colour images. *Lecture Notes in Computer Science 3656* (2005), 667–675.

[17] FARLEX. *Colour-blindness*. http://www.thefreedictionary.com.

[18] FLÓREZ-REVUELTA, F. Ordering of the RGB space with a growing self-organizing network. Application to color mathematical morphology. *Lecture Notes in Computer Science 3696* (2005), 385–390.

[19] FORD, A., AND ROBERTS, A. *Colour space conversions*. http://www.poynton.com.

[20] FREEMAN, W., JONES, T., AND PASZTOR, E. Example-based super-resolution. *IEEE Computer Graphics and Applications 22*, 2 (2002), 56–65.

[21] GOGUEN, J. $\mathcal{L}$-fuzzy sets. *Journal of Mathematical Analysis and Applications 18* (1967), 145–174.

[22] HANBURY, A., AND SERRA, J. Mathematical morphology in the HLS colour space. In *Proceedings of 12th British Machine Vision Conference BMV 2001* (United Kingdom, 2001), pp. II – 451–460.

[23] HANBURY, A., AND SERRA, J. Mathematical morphology in the CIELAB space. *Image Analysis and Stereology 21*, 3 (2002), 201–206.

[24] HANBURY, A. G., AND SERRA, J. Morphological operators on the unit circle. *IEEE Transactions on Image Processing 10*, 12 (2001), 1842–1850.

[25] HEIJMANS, H. J. A. M. *Morphological Image Operators*. Academic Press, USA, 1994.

[26] HEIJMANS, H. J. A. M., AND RONSE, C. The algebraic basis of mathematical morphology, part 1: Dilations and erosions. *Computer Vision, Graphics and Image Processing 50*, 3 (1990), 245–295.

[27] HONDA, H., HASEYAMA, M., AND KITAJIMA, H. Fractal interpolation for natural images. In *Proceedings of IEEE International Conference on Image Processing ICIP 1999* (Japan, October 1999), pp. 657–661.

[28] KERRE, E. E. *Fuzzy Sets and Approximate Reasoning*. Xian Jiaotong University Press, 1998.

[29] LEDDA, A., LUONG, H., DE WITTE, V., PHILIPS, W., AND KERRE, E. E. Image interpolation using mathematical morphology. In *Sixth FirW PhD Symposium, Ghent, University* (Belgium, 2005), p. 94.

[30] LEDDA, A., LUONG, H., PHILIPS, W., DE WITTE, V., AND KERRE, E. E. Greyscale image interpolation using mathematical morphology. *Lecture Notes in Computer Science 4179* (2006), 78–90.

[31] LEDDA, A., LUONG, H., PHILIPS, W., DE WITTE, V., AND KERRE, E. E. Image interpolation using mathematical morphology. In *Proceedings of Second International Conference on Document Image Analysis for Libraries DIAL 2006* (France, April 2006), pp. 358–367.

[32] LEHMANN, T., GÖNNER, C., AND SPITZER, K. Survey: Interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging 18*, 11 (1999), 1049–1075.

[33] LI, J., AND LI, Y. Multivariate mathematical morphology based on principal component analysis: initial results in building extraction. *International Archives for Photogrammetry, Remote Sensing and Spatial Information Sciences 35*, B7 (2004), 1168–1173.

[34] LI, X., AND ORCHARD, M. New edge-directed interpolation. *IEEE Transactions on Image Processing 10*, 10 (2001), 1521–1527.

[35] LOUVERDIS, G., AND ANDREADIS, I. Design and implementation of a fuzzy hardware structure for morphological color image processing. *IEEE Transactions on Circuits and Systems for Video Technology 13*, 3 (2003), 277–288.

[36] LOUVERDIS, G., ANDREADIS, I., AND TSALIDES, P. New fuzzy model for morphological color image processing. In *Proceedings of IEEE Vision, image and signal processing* (2002), pp. 129–139.

[37] LOUVERDIS, G., VARDAVOULIA, M., ANDREADIS, I., AND TSALIDES, P. A new approach to morphological color image processing. *Pattern Recognition 35*, 8 (2002), 1733–1741.

[38] LUONG, H., DE SMET, P., AND PHILIPS, W. Image interpolation using constrained adaptive contrast enhancement techniques. In *Proceedings of IEEE International Conference on Image Processing ICIP 2005* (Italy, September 2005), pp. 998–1001.

[39] M. I. VARDAVOULIA, I. A., AND TSALIDES, P. Vector ordering and morphological operations for colour image processing: fundamentals and applications. *Pattern Analysis & Applications 5* (2002), 271–287.

[40] MACADAM, D. L. Visual sensitivities to color differences in daylight. *Journal of the Optical Society of America 32*, 5 (1942), 247–274.

[41] MATHERON, G., AND SERRA, J. *The birth of mathematical morphology*. http://cmm.ensmp.fr/∼serra/communications_pdf/C-72.pdf, 1998.

[42] MORSE, B. S., AND SCHWARTZWALD, D. Isophote-based interpolation. In *Proceedings of IEEE International Conference on Image Processing ICIP 1998* (USA, October 1998), pp. 227–231.

[43] NACHTEGAEL, M. *Vaagmorfologische en vaaglogische filtertechnieken in beeldverwerking*. PhD Thesis, Ghent University, Belgium, 2002.

[44] NACHTEGAEL, M., AND KERRE, E. E. Connections between binary, gray-scale and fuzzy mathematical morphologies. *Fuzzy Sets and Systems 124*, 1 (2001), 73–85.

[45] NACHTEGAEL, M., SCHULTE, S., VAN DER WEKEN, D., DE WITTE, V., AND KERRE, E. E. Fuzzy filters for noise reduction: the case of impulse noise. In *Proceedings of Joint 2nd International Conference on Soft Computing and Intelligent Systems and 5th International Symposium on Advanced Intelligent Systems SCIS and ISIS 2004* (Japan, September 2004), pp. CD–ROM.

[46] NACHTEGAEL, M., SCHULTE, S., VAN DER WEKEN, D., DE WITTE, V., AND KERRE, E. E. Do fuzzy techniques offer an added value for noise reduction in images? *Lecture Notes in Computer Science 3708* (2005), 658–665.

[47] NACHTEGAEL, M., SCHULTE, S., VAN DER WEKEN, D., DE WITTE, V., AND KERRE, E. E. Fuzzy filters for noise reduction: the case of gaussian noise. In *Proceedings of 14th IEEE International Conference on Fuzzy Systems FUZZ-IEEE 2005* (Nevada, 2005), pp. 201–206.

[48] NACHTEGAEL, M., SCHULTE, S., VAN DER WEKEN, D., DE WITTE, V., AND KERRE, E. E. Gaussian noise reduction in greyscale images. *International Journal of Intelligent Systems Technologies and Applications 1*, 3 (2006), 211–233.

[49] NACHTEGAEL, M., VAN DER WEKEN, D., SCHULTE, S., AND DE WITTE, V. Fuzzy techniques in image processing at Ghent University. *Journal of Intelligent and Fuzzy Systems 16*, 4 (2005), 281–287.

[50] NACHTEGAEL, M., VAN DER WEKEN, D., SCHULTE, S., DE WITTE, V., AND KERRE, E. E. Fuzzy mathematical morphology: current status and required developments. In *Proceedings of 6th International Symposium on Advanced Intelligent Systems ISIS 2005* (Korea, 2005), pp. 429–434.

[51] NATIONAL EYE INSTITUUT. *Diagram of the eye*. http://www.nei.nih.gov.

[52] PETERS, R. A. Mathematical morphology for angle-valued images. In *Proceedings of Nonlinear Image Processing VIII SPIE 3026* (USA, 1997), pp. 84–94.

[53] PHILIPS, W. *Image Processing*. http://telin.ugent.be/ philips/beeldv/.

[54] POWER RETOUCHE. *Staircasing filter*. http://www.powerretouche.com/.

[55] POYNTON, C. *Frequently asked questions about color and Frequently asked questions about gamma*. http://www.poynton.com.

[56] RONSE, C. Why mathematical morphology needs complete lattices. *Signal Processing 21*, 2 (1990), 129–154.

[57] RONSE, C., AND HEIJMANS, H. J. A. M. The algebraic basis of mathematical morphology, part 2: Openings and closings. *Computer Vision, Graphics and Image Processing 54*, 1 (1991), 74–97.

[58] SANGWINE, S. J., AND HORNE, R. E. N. *The Colour Image Processing Handbook*. Chapman & Hall, 1998.

[59] SARTOR, L. J., AND WEEKS, A. R. Morphological operations on color images. *Journal of Electronic Imaging 10*, 2 (2001), 548–559.

[60] SCHULTE, S., DE WITTE, V., AND KERRE, E. E. A fuzzy noise reduction method for color images. *IEEE Transactions on Image Processing* (2007), in press.

[61] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. A new fuzzy multi-channel filter for the reduction of impulse noise. *Lecture Notes in Computer Science 3522* (2005), 368–375.

[62] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. A novel histogram based fuzzy impulse noise restoration method for colour images. *Lecture Notes in Computer Science 3708* (2005), 626–633.

[63] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. A new fuzzy filter for the reduction of randomly valued impulse noise. In *Proceedings of IEEE International Conference on Image Processing ICIP 2006* (USA, 2006), pp. 1809–1812.

[64] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. Fuzzy two-step filter for impulse noise reduction from color images. *IEEE Transactions on Image Processing 15*, 11 (2006), 3567–3578.

[65] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. Fuzzy random impulse noise reduction method. *Fuzzy Sets and Systems 158*, 3 (2007), 270–283.

[66] SCHULTE, S., DE WITTE, V., NACHTEGAEL, M., VAN DER WEKEN, D., AND KERRE, E. E. Histogram-based fuzzy colour filter for image restoration. *Image and Vision Computing* (2007), in press.

[67] SCHULTE, S., NACHTEGAEL, M., DE WITTE, V., VAN DER WEKEN, D., AND KERRE, E. E. A new two step color filter for impulse noise. In *Proceedings of the East West Fuzzy Colloquium 2004 (11th Zittau Fuzzy Colloquium)* (Germany, 2004), pp. 185–192.

[68] SCHULTE, S., NACHTEGAEL, M., DE WITTE, V., VAN DER WEKEN, D., AND KERRE, E. E. A fuzzy impulse noise detection and reduction method. *IEEE Transactions on Image Processing 15*, 5 (2006), 1153–1162.

[69] SCHULTE, S., NACHTEGAEL, M., DE WITTE, V., VAN DER WEKEN, D., AND KERRE, E. E. *Fuzzy Impulse Noise Reduction Methods for Color Images*. Springer, 2006, ch. Computational intelligence, theory and applications, pp. 711–720.

[70] SCRUGGS, J. *Color theory*. http://www.bway.net/ jscruggs/index3.html.

[71] SERRA, J., AND SOILLE, P. *Mathematical Morphology and Its Applications to Image Processing*. Kluwer Academic Publishers, 1994.

[72] SHARMA, G. *Digital Color Imaging Handbook*. CRC-Press, England, 2003.

[73] SHARMA, G., AND TRUSSELL, H. J. Digital color imaging. *IEEE Transactions on Image Processing 6*, 7 (1997), 901–932.

[74] SOILLE, P. *Morphological Image Analysis: Principles and Applications*. Springer-Verlag, 1999.

[75] SPECIALCHEM. *Color centre*. http://www.specialchem4coatings.com/tc/color-handbook.

[76] STEPIN, M. *Hq 3x magnification filter*. http://www.hiend3d.com/hq3x.html.

[77] TALBOT, H., EVANS, C., AND JONES, R. *Mathematical Morphology and Its Applications to Image and Signal Processing*. Kluwer Academic Press, 1998, ch. Complete ordering and multivariate mathematical morphology, pp. 27–34.

[78] TSCHUMPERLÉ, D. *PDE's based regularization of multivalued images and applications*. PhD Thesis, Université de Nice, France, 2002.

[79] UNIVERSITY OF VIRGINIA, DEPARTMENT OF ASTRONOMY. *The electromagnetic spectrum*. www.astro.virginia.edu.

[80] UNSER, M., ALDROUBI, A., AND EDEN, M. B-spline signal processing: Part 1 - Theory. *IEEE Transactions on Signal Processing 41*, 2 (1993), 821–833.

[81] UNSER, M., ALDROUBI, A., AND EDEN, M. B-spline signal processing: Part 2 - Efficient design and applications. *IEEE Transactions on Signal Processing 41*, 2 (1993), 834–848.

[82] VAN DER WEKEN, D., DE WITTE, V., NACHTEGAEL, M., SCHULTE, S., AND KERRE, E. E. Colour image comparison using vector operators. In *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce CIMCA-IAWTIC 2005* (Austria, 2005), pp. 295–300.

[83] VAN DER WEKEN, D., DE WITTE, V., NACHTEGAEL, M., SCHULTE, S., AND KERRE, E. E. A component-based and vector-based approach for the construction of quality measures for colour images. In *Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU 2006* (France, 2006), pp. CD–ROM.

[84] VAN DER WEKEN, D., DE WITTE, V., NACHTEGAEL, M., SCHULTE, S., AND KERRE, E. E. Fuzzy similarity measures for colour images. In *Proceedings of IEEE Conference on Cybernetics and Intelligent Systems 2006* (Thailand, 2006), pp. 1–6.

[85] VAN DER WEKEN, D., NACHTEGAEL, M., DE WITTE, V., SCHULTE, S., AND KERRE, E. E. A survey on the use and the construction of fuzzy similarity measures in image processing. In *Proceedings of IEEE International Conference on Computational Intelligence for Measurement Systems and Applications CIMSA 2005* (Italy, 2005), pp. 187–192.

[86] VAN DER WEKEN, D., NACHTEGAEL, M., DE WITTE, V., SCHULTE, S., AND KERRE, E. E. Constructing similarity measures for colour images. In *Proceedings of the International Fuzzy Systems Association World Congress on Fuzzy Logic, Soft Computing and Computational Intelligence Theories and Applications IFSA 2005* (China, 2005), pp. 1020–1025.

[87] WIKIPEDIA. *Color*. http://en.wikipedia.org/wiki/Category:Color.

[88] ZADEH, L. A. Fuzzy sets. *Information and Control 8* (1965), 338–353.