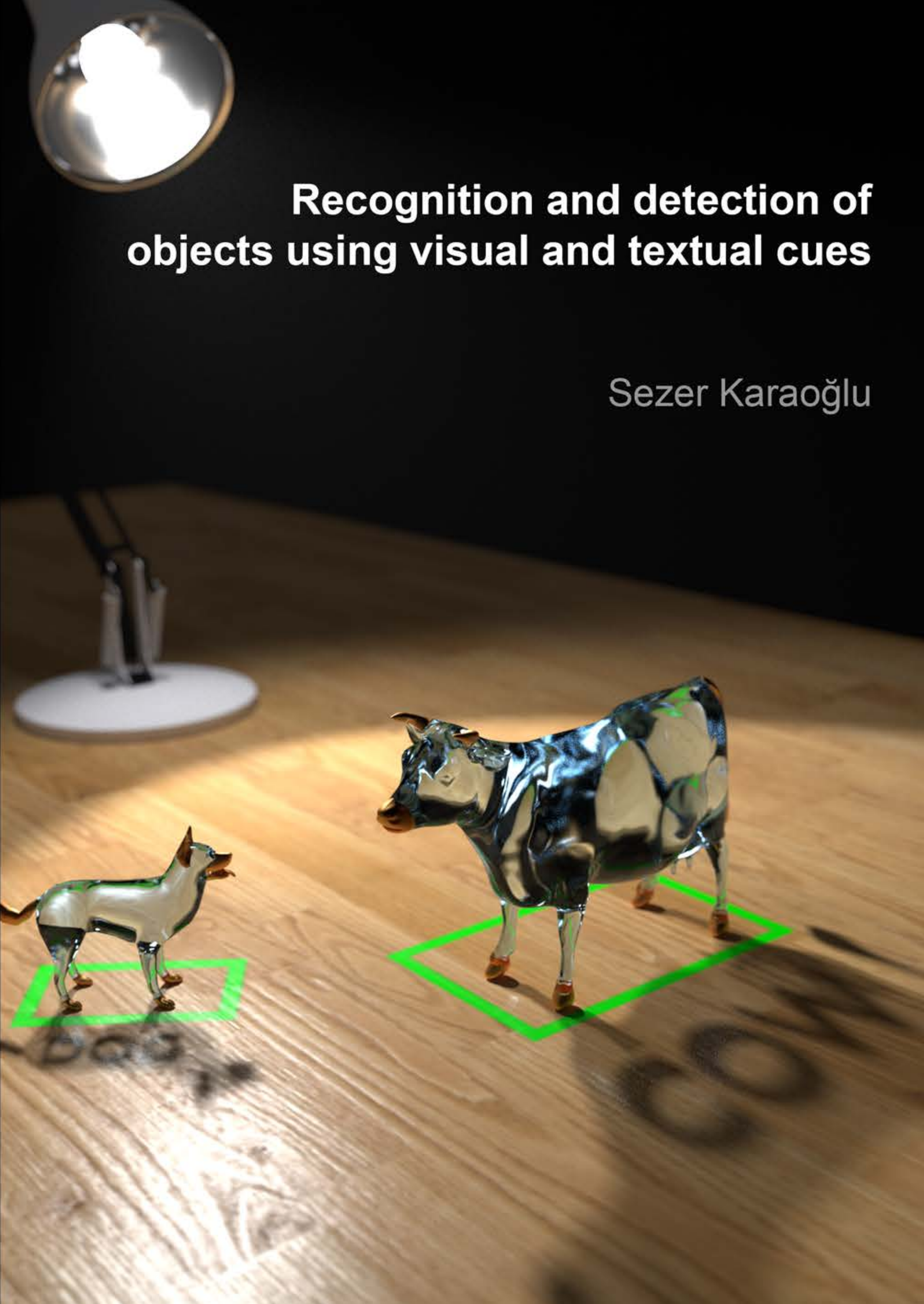
Recognition and detection of objects using visual and textual cues

Karaoğlu, S.

Recognition and detection of objects using visual and textual cues

Sezer Karaoğlu



Recognition and Detection of Objects Using Visual and Textual Cues

Sezer Karaoğlu

This book was typeset by the author using LaTeX 2_ε.

Printing: Off Page, Amsterdam

ISBN 978-94-6182-745-6

Recognition and Detection of Objects Using Visual and Textual Cues

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. K. I. J. Maex
ten overstaan van een door het college voor promoties
ingestelde commissie,
in het openbaar te verdedigen in de Aula der Universiteit
op vrijdag 2 december 2016, te 11.00 uur

door

Sezer Karaoğlu

geboren te Tarsus, Turkije

Promotiecommissie

Promotor:	Prof. dr. T. Gevers	University of Amsterdam
Co-promotor:	Prof. dr. ir. A.W.M. Smeulders	University of Amsterdam
Overige Leden:	Prof. dr. A. Trémeau	Université Jean Monnet
	Prof. dr. A. Hanjalić	Delft University of Technology
	Prof. dr. M. de Rijke	University of Amsterdam
	Dr. C. G. M. Snoek	University of Amsterdam
	Dr. S. Rudinac	University of Amsterdam

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

This research was supported by the Dutch national program COMMIT.

This work was carried out in the ASCI graduate school at University of Amsterdam.
ASCI dissertation series number 365.

UNIVERSITEIT VAN AMSTERDAM

Anne ve Babama.

Contents

1

Introduction

The history of image making and understanding came a long way from ancient times. It started nearly around 40.000 years ago, when the first paintings were practiced on cave walls. The early messages were conveyed via objects followed by time and action. The artists initially drew animals (objects) such as bulls, horses and bisons which played an essential role (food) in their lives. The initial drawings were mainly outlines of the objects. Specifically, objects were represented using boundaries with emphasizes on distinctive object parts (e.g. horn of a bison). Then, the artists start to add more details about the object (e.g. fur color of a bison). Later, the artists wanted to extend the messages by adding high-level meanings to these drawings. For example, to indicate the time of the day (e.g. mid-day), lighting effects are used (e.g. shadow under a bull), see Fig. 1.1. The posture of the animals was used to make them appear to be either alive or dead. To express motion, for example, bulls with eight legs were painted to indicate running. There has been a long debate about the purpose of these paintings. One of the most popular hypotheses suggests that the depictions were used to transfer knowledge, habits, experiences and culture to the next generation [120, 154]. According to this hypothesis, our ancestors wanted to help the next generations of human kind to survive by sharing their valuable knowledge. Beyond images themselves, understanding these images also carried an important role on the continuation of the human kind. Humans must have assessed the message carried by these images. This demonstrates the ability of humans to understand what they see. In fact, humans have a very well developed vision system. We can extract low-level information (e.g. color, lines and texture) from an image and understand the high-level message it conveys (e.g. recognizing a bull only from an outline). Therefore, even today, it is not so hard to interpret what these drawings convey when we look at them.

Although the urge to tell stories did not change, image making has been evolving by the years. Nowadays, not only the artists (e.g. cave walls) but also anyone with a mobile phone can create pictures. In fact, if you want, you can record and share every single moment in your life. Every minute, millions of media elements (e.g. pictures and videos) are uploaded to the Internet. Understanding and archiving this enormous amount of data is only possible by automatization.

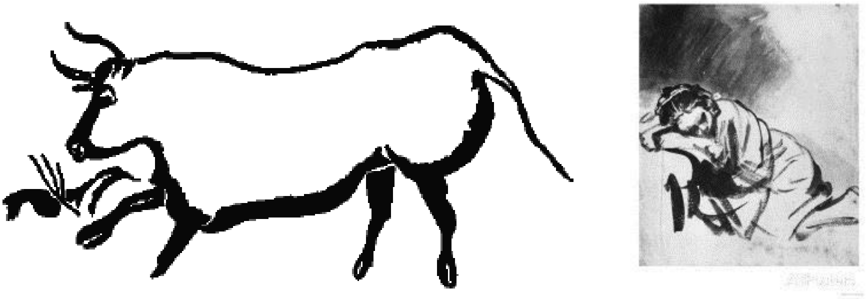


Figure 1.1: In the 23,000-year old picture from Lascaux, the artist conveys that it is midday by letting the light come from above, where Rembrandt indicates the woman is resting in the evening hour.

Automatization requires intelligent algorithms that interpret and understand the world like humans do. To achieve this, we need to teach machines how to see. In this thesis, we explore images in terms of fundamental elements for what a person needs to see (i.e. light, object and high level semantic interpretation).

The human vision system is the result of complex interactions between light, objects, eyes and brain. For example, color is not an inherent characteristic of objects [131]. What we observe as color is the result of object surfaces reflecting and absorbing different wavelengths which are part of the electromagnetic spectrum. Hence, from a physics-based perspective, light is the only quantity that we are able to observe. Light, emitted by a source (e.g. lamp), is reflected by object surfaces and stimulates the photoreceptors of our eyes. Then, photoreceptors convert the incoming light into signals which are transferred to our brain. Subsequently, our brain deciphers the information to detect low-level cues such as color, edges and texture of objects and scenes. Then, these low-level cues are grouped together to form high-level cues for semantic interpretation.

As sight starts with light, so does this thesis. As discussed, light is an inevitable factor to form images. Besides the importance on forming an image, light has also attracted attention of research to extract other low-level (e.g. surface structure [61, 75, 156, 196]) and high-level (e.g. material property [94]) cues. To this end, we can understand if a surface is matte/glossy which may also interpret softness/hardness (e.g. a matte plastic or a shiny metal) of the surface. Or we can understand the underlying geometry of objects (due to shading cues). To exploit the interaction between light and objects, it is important to derive the position of the light source with respect to the objects. Therefore, the second chapter of this thesis focuses on estimating the location of the light source (i.e. light source position). To address this problem we consider a well established technique which relies on Lambert's law. Lambert's law states that the quantity of light absorbed is proportional to the angle between the light source and the surface normal direction. Consequently, most of the algorithms aim to infer the light source position (LSP) using pixel intensities corresponding with surface normal [180, 5, 198, 148, 91, 106, 8]. This is straight-forward as long as the Lambert's law holds, i.e. matte surfaces. However, a glossy surface is prone to (specular) highlights and material-to-material inter-reflections which are not considered by Lambert's law. Moreover, due to imperfections of the recording devices, the surface normal may become noisy on rough, crinkled and grained surfaces. These types of surfaces may negatively influence the LSP estimation. Therefore, we pose the following question:

- *Can we assign importance to surfaces based on their surface attributes to improve the estimation of the light source position?*

To address this question, in Chapter 2, we propose a method which exploits the influence of various surface attributes for LSP estimation. More importance is rendered to those image surface segments which have suitable photometric and geometric surface attributes. We study whether weighting image surface segments based on their attributes may outperform the state-of-the-art methods in which the image surface segments are considered to equally contribute.

After light is converted into signals and transferred to our brain, the initial stage of visual processing consists of determining discontinuities in color, depth and texture. A number of these discontinuities capture distinctive information (patterns) about the objects. The human visual system makes use of these patterns to recognize objects even in the situation where there are large variations between object appearances and imaging conditions. More specifically, objects can be located at any location on the retina (position), at any distance (scale), for different views (pose), and for different lighting conditions (Illumination) [24]. Moreover, instances of the same object class should be considered as the same category (intra-class) independent of the different patterns introduced by the varying imaging conditions. The main question is how to establish the equivalence between all of these patterns. Many successful works have proposed invariant representations to achieve robust recognition over a large range of imaging and viewing conditions [7, 48, 49, 98, 115, 194]. A full invariant representation, unfortunately, leads to a decrease in discriminative power [172]. This drawback is due to distinguishing transformations that a full invariant representation cannot capture. For example, under rotational invariance a "6" is identical to a "9". The current invariant representations do not allow having robustness for only certain degree of freedom. These rigid properties of current invariants play a central role in the trade-off between invariance and discriminative power. Subsequently, in this thesis, we try to find an answer to our second research question:

- *Is it possible to enable a degree of robustness to local image descriptors instead of on/off invariant property for more discriminative object recognition?*

To this end, in Chapter 3, we propose to replace these binary on/off invariants by steering the robustness to a limited range of changes. When the robustness is restricted, the descriptors are expected to be more discriminative.

Low-level visual cues are not sufficient to understand the deeper meaning of images. It is important to extract higher-level cues as well. In the human brain, the initial step towards is to combine low-level cues to recognize objects. Therefore, we aim for an algorithm to recognize objects in the scene. Object recognition algorithms have shown significant progress on classifying "distinct" object categories such as horses, bicycles and cars [34]. These algorithms mostly rely on appearance cues (e.g. color, texture and shape) [21, 168]. However, these visual cues may not be sufficient enough to distinguish categories of objects that only slightly differ in appearance (e.g. fine-grained classification and instance search). Complementary cues are essential to distinguish visually similar object categories. The requirement of complementary source of information brings out a new research question:

- *Is scene-text useful for image classification?*

In Chapter 4, we make use of the domain specific knowledge of buildings and logos. Texts in natural scenes (i.e., outside a document) typically add meaning to an object, location or scene. For example, text adds identification on the brand or type of a product, it specifies which buildings serve food (e.g. steakhouse) or drinks (e.g. tea, coffee), and what kind of service is provided (e.g. dry-cleaning, repair). The mere presence of text, its words and their meaning are closely related to the semantics of the scene. To this end, in Chapter 4, we propose to exploit this semantic relationship between text and the scene to improve automatic image classification and retrieval where visual cues may not prove sufficient. We propose a generic, efficient and fully unsupervised character detection algorithm. Detected character regions are processed by a state-of-the-art character recognition algorithm [69]. Then, bi- and trigrams (textual cues) are formed between the recognized characters by considering spatial pairwise constraints. Extracted textual and visual cues are combined for fine-grained classification and image retrieval.

In Chapter 4, we consider textual cues encoded at character level for fine-grained classification and image retrieval. However, characters may not fully convey the rich semantics given by the text. Accordingly, the next research question is investigated:

- *Can we achieve a better image classification and retrieval by using word-level textual cue encoding rather than character-level textual cue encoding?*

To this end, in Chapter 5, we propose to encode textual cues at word-level. Moreover, it is widely acknowledged in text detection literature that high f-score in word detection is more important than high recall. However, when a word in an image is not detected or localized incorrectly, it is not possible to identify it. Therefore, in Chapter 5, we also investigate the following research question:

- *Is high recall in word detection more important than high f-score for fine-grained classification and image retrieval?*

The message carried by recognized objects may also be limited. Therefore, it is essential to probe deeper by exploring other cues (e.g. object location). Locating an object in an image may reveal the information based on object spatial relations. For instance, a person on top of a bike may represent cycling, a person with a gun in his hand may raise alert or shadows under a cow may represent mid-day as in cave paintings. To exploit such high-level semantics, we investigate where the object is (i.e. object detection) in Chapter 6. Many object detection algorithms have been proposed in the literature [19, 41, 52, 58, 86, 110, 167, 173]. Although these detection algorithms are successful for common object appearances and imaging conditions [60], their design properties (e.g. search strategy, features, and model presentation) influence the robustness of these methods to varying imaging conditions (e.g. occlusion, clutter, unusual views, and

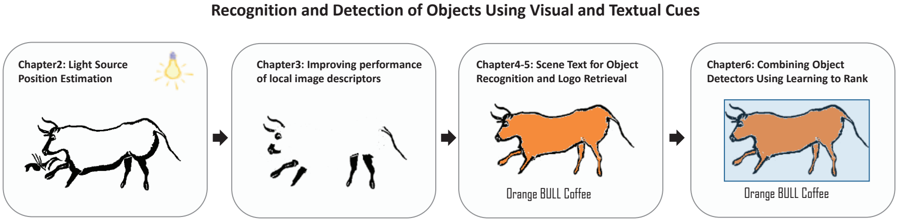

Recognition and Detection of Objects Using Visual and Textual Cues
Chapter2: Light Source Position Estimation
Chapter3: Improving performance of local image descriptors
Chapter4-5: Scene Text for Object Recognition and Logo Retrieval
Orange BULL Coffee
Chapter6: Combining Object Detectors Using Learning to Rank
Orange BULL Coffee

Figure 1.2: The thesis is structured as follows: Chapter 2 of this thesis focuses on light source position estimation. In Chapter 3, we propose a method to improve matching performance of local image descriptors. In Chapter 4 and 5, we combine textual and visual cues to improve fine-grained place of business classification and logo retrieval. Finally, in Chapter 6, we propose a method to combine state-of-the-art object detectors to improve single detector performances.

object size). As a consequence, no detection algorithm can be considered as universal. With the large variety of available methods, the question is:

- *How can we combine state-of-the-art object detectors to preserve their strengths while reducing their limitations and assumptions?*

To this end, in Chapter 6, we consider a rank learning approach to combine object detection methods. The proposed framework combines detections (detector outputs which consist of a classifier score and bounding box locations) of various well-known object detectors including DPM [41], CN [86], EES [110] and RCNN [52]. The aim of the proposed method is to provide advantages over the state-of-the-art object detection algorithms. (1) Missed detections (false negatives) of single detectors are compensated by combining detections of different detectors (due to complementary detections). (2) Detections are re-ranked by using information gathered by other detectors (due to common detections). True detections (true positives) of each detector are rewarded and false detections (false positives) of each detector are penalized within the learning to rank framework. (3) The combined list maintains the strengths of the detectors. Therefore, it is more robust than each individual detector for varying imaging conditions.

This thesis focuses on various aspects of image understanding, from improving matching performance of local image descriptors (low-level) to recognizing and detecting objects in a scene (high-level). We consider each research question addressed in this thesis as a piece of a complex image understanding puzzle. Solving unique challenges in each piece of the puzzle may allow us to reveal the overall picture and acquire in depth image understanding. The outline of the thesis is depicted in Fig. 1.2.

1.1 Origins

This thesis is constructed based on seven research papers [79, 77, 82, 83, 81, 80, 78]. For each chapter of this thesis, we list the publications and specify the role of each co-author below:

Chapter 2 is based on *Point Light Position Estimation from RGB-D Image Sequences by Learning Surface Attributes* under review for publication in IEEE Transactions on Image Processing [79] by Karaoglu, Liu, Gevers and Smeulders. Theories and algorithms were led by Karaoglu, with contributions from the co-authors. The experiments and the analysis were performed by Karaoglu, Liu contributed to the experiments. All authors contributed to the text.

Chapter 3 is based on *Per-patch Metric Learning for Robust Image Matching* at ICIP'15 [77] by Karaoglu, Everts, Gemert and Gevers. Karaoglu led the development of the algorithms and experiments, with contributions from the co-authors. Karaoglu initiated writing the paper and all co-authors contributed to the text.

Chapter 4 is based on *Object Reading: Text Recognition for Object Recognition* at ECCV-W'12 [82] by Karaoglu, Gemert and Gevers and *Con-Text: Text Detection Using Background Connectivity for Fine-Grained Object Classification* at ACM-MM'13 [83] by Karaoglu, Gemert and Gevers. Extended and combined version of these papers *Con-Text: Text Detection for Fine-grained Classification and Logo Retrieval* is under review in IEEE Transactions on Image Processing [81] by Karaoglu, Tao, Gemert and Gevers. Karaoglu developed all the algorithms and performed all the experiments. The co-authors contributed to the experiments. The paper was written by Karaoglu. The co-authors contributed to the text.

Chapter 5 is based on *Words Matter: Scene Text for Image Classification and Retrieval* under review for publication in IEEE Transactions on Multimedia [80] by Karaoglu, Tao, Gevers and Smeulders. Karaoglu and Tao equally contributed to this work. Theories and algorithms were developed together. Karaoglu focused on implementing textual cue extraction, whereas Tao focused on implementing visual cue extraction for fine-grained classification and logo retrieval tasks. The experiments, the analysis and paper writing were performed by Karaoglu and Tao. All authors contributed to the text.

Chapter 6 is based on *Detect2Rank: Combining Object Detectors Using Learning to Rank* published in IEEE Transactions on Image Processing [78] by Karaoglu, Liu and Gevers. Karaoglu developed all the algorithms and performed all the experiments, with contributions from Gevers. The paper was written by Karaoglu. The co-authors contributed to the text.

2

Light Source Position Estimation*

2.1 Introduction

Images are the result of complex interactions between the light source, objects and recording devices. Being the creator of the image before anything else, the light source is often ignored as an important cue to its understanding. The light source may reveal low-level (e.g. surface structure [61, 156, 75, 196]) and high-level (e.g. material property [94]) information. Such information is used by humans in their daily activities. For instance, we can distinguish whether a surface is matte/glossy and what material the surface is made of (e.g. a matte plastic or a shiny metal) [38]. Or we can interpret the underlying geometry of objects (due to shading cues). The interaction between the light source and objects is also used in visual art where artists exploit the characteristics of light in different ways. Specular reflections in an human eye give the lively twinkle but are in fact a direct reflection of the light source. Shading indicates the curvature of the body and reveals collimation and direction of the light. Spotlight steers salience, and backlight in art photography renders the subject radiant. We consider that light source is an inevitable factor of forming and understanding images. Therefore, in this chapter, we focus on detecting the light source position.

Light Source Position (LSP) estimation has caught attention in references such as [180, 5, 198, 148, 91, 106]. Most of these algorithms are based on Lambert's law, assuming that the pixel intensity is proportional to the angle between the light and surface (normal) direction. LSP estimation algorithms infer the position by assuming certain 3D-shapes of objects in the scene [180] (to obtain surface normals). Often these assumptions fail, and hence the applicability of these methods is limited.

A recent approach is to use low cost RGB-D cameras (e.g. Kinect and Asus Xtion) as they

[*] S. Karaoglu, Y. Liu, T. Gevers and A. W. M. Smeulders, "Point Light Position Estimation from RGB-D Image Sequences by Learning Surface Attributes," Under revision in IEEE Transactions on Image Processing (TIP), 2016.

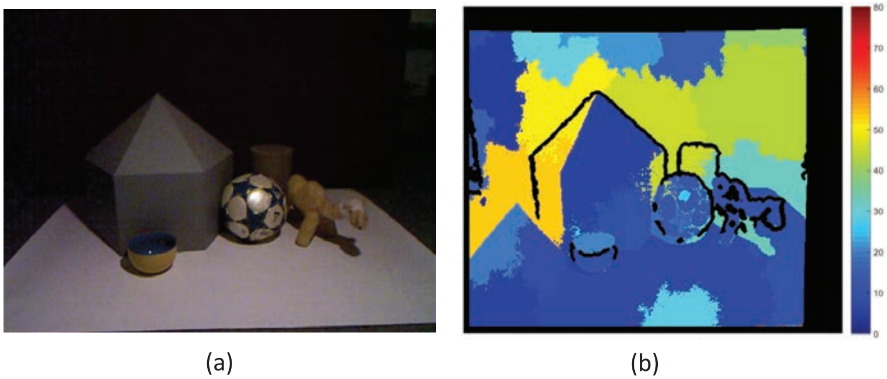


Figure 2.1: (Best viewed in color) (a) An image sample taken from [8]. (b) Image surface segments produced by the mean-shift algorithm and their individual angular errors. The color represents the angular error of each image surface segment. The corresponding angular error for each color is at the colorbar next to each plot. Dark blue regions represent smaller angular errors whereas dark red regions represent larger angular errors. The figure illustrates that curved (such as ball and bowl) and less textured (such as table and toy castle) image surface segments have lower LSP estimation errors. Moreover, regions with shadows or highlights have difficulties to estimate LSP.

acquire color images with their depth in real-time. The use of RGB-D cameras alleviates the requirement of assuming certain object shapes, because the surface normals can be readily computed [8, 129, 72]. Assuming Lambert's law, the light source position can be estimated using the surface normals and pixel intensities. In particular, LSP is obtained by minimizing the residuals between the re-rendered (scene generated using a hypothesized light source position) and the original scene. This is straight-forward as long as Lambert's law holds, such as for matte surfaces. However, a glossy surface is prone to specular highlights and material-to-material inter-reflections which are not considered by Lambert's law. Moreover, due to imperfections of recording devices, the surface normals may be noisy on rough, crinkled and grained surfaces. Hence, these type of surfaces may negatively influence the LSP estimation. In Fig. 2.1, we estimate the LSP for each image surface segment (obtained by color-based mean-shift algorithm). We measure the angular error between the estimated LSP and ground-truth LSP. The figure illustrates that different image surface segments have varying LSP estimation errors.

In this chapter, we propose a method which exploits the influence of various surface attributes for LSP estimation. First, surface attributes are computed from image surface segments. These attributes are used in a supervised learning scheme to rank the suitability of each surface for proper LSP estimation. Higher importance is assigned to image surface segments which have proper photometric (i.e. Lambertian reflectance) and geometric surface attributes. This is an advantage over the state-of-the-art methods [8, 72] which assume equal importance to all surfaces. To improve the performance further, we introduce temporal constraints, extending the method to

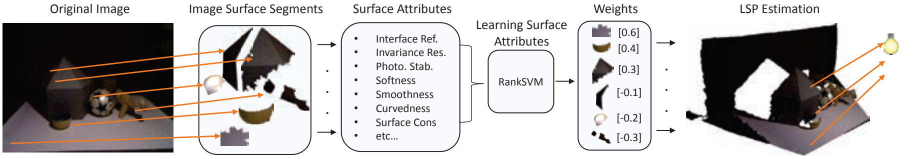


Figure 2.2: The flow of the proposed approach. First, the image is divided into image surface segments. Surface attributes (e.g. glossy, matte, highlight, curved etc) are extracted from these regions aiming to rank them based on their suitability for proper LSP estimation. The proposed method assigns more importance to image surface segments which are more suitable for LSP estimation. This is in contrast to the state-of-the-art methods [8, 129] which assume equal importance to all image surface segments. Moreover, to improve the performance further, we introduce temporal constraints (for video sequences).

video sequences. For static objects, it is assumed that the light source position does not change during the recording of the video. We derive a temporal constraint by estimating the camera pose. The camera pose is used to estimate a global LSP which minimizes the residuals between the re-rendered video frames and the original video frames. See Fig. 6.1 for the outline of the proposed method. The proposed method is tested on two different datasets (*Boom13* [8] and our newly collected video dataset). Experiments on these datasets show that surface weighting provides a significant improvement over the state-of-the-art methods.

The chapter has the following contributions. First, surface attributes are differentiated according to their importance for LSP estimation. Second, in contrast to state-of-the-art methods, which declare all surface contributions as equally important, we derive weights for individual image surface segments based on their suitability. Third, a geometry-based initialization is proposed to make the light source LSP estimation specific to the underlying image and to ensure fast convergence. Fourth, for videos, we introduce a global consistency term to constrain the light source positioning by estimating the camera pose. Finally, we prove the viability of the method on a new (video) dataset, to be made publicly available.

2.2 Related Work

Visual Cues. In general, light source positioning algorithms assume certain $3D$ object shapes or known objects in the scene. For instance, [148] assumes that the position of objects in the scene is known and uses cast shadows to estimate the light source direction. [113] uses a fisheye camera to recover the position of the light source for indoor scenes. Others, e.g. [106], use GPS and compass information to determine the sun position in outdoor scenes. In [91], various cues are extracted from the sky, vertical surfaces and the ground to estimate the direction of the sun. Although these methods may be suited to estimate the light source direction, they impose strong assumptions on the imaging conditions.

Depth Cues. More recently, a number of methods have been proposed that use depth information.

For instance, [129] decompose the image into specular, diffuse and albedo layers. The specular and diffuse layers are used to constrain the difference between the re-rendered and original image. However, detecting specular parts in a 2D still image is a difficult problem. Moreover, each surface segment is considered to equally contribute to LSP estimation. [8] assumes that image surface segments with the same color have the same albedo. The authors re-render the scene from a hypothesized LSP while minimizing the error between the synthesized and original image.

In contrast to the previous methods, we propose to learn the suitability of surfaces to estimate the LSP based on their surface attributes. We assign weights to the image surface segments rather than treating them all equally.

2.3 LSP Estimation Using Surface Suitability

According to Lambert's law, an intensity pixel value I can be modeled by:

$$I(u) = \rho(u) min(\vec{n}(u)(\frac{L - p(u)}{\|L - p(u)\|})^t \imath, 0) \ , \tag{2.1}$$

where t represents the transpose. The intensity value I at pixel u, depends on the surface albedo ρ, the surface normal $\vec{n}$, the light source direction and the intensity $\imath$ of the light. The light source direction is defined as the direction between the light source position L and the point p in $3D$ coordinates. From the RGB-D sensor, both the intensity I and the depth images are provided. Further, $\vec{n}$ is computed from the depth image using [147]. For LSP estimation, the albedo ρ and the light intensity $\imath$ are unknown. Hence, it is necessary to estimate ρ and $\imath$. Image surface segments are generated by using the (color-based) mean-shift algorithm. It is assumed that image surface segments have uniform albedo [8, 72, 129]. Thus, ρ does not change within an image surface segment. Moreover, it is assumed in [8] that $\imath$ remains constant because the light source distance does not vary significantly over neighboring pixels (image surface segment). We also do not use the inverse square law to adjust $\imath$. We set $\imath$ to be 1. Under these assumptions, the LSP is estimated by minimizing the error between the reconstructed and original image surface segments based on the hypothesized LSP.

2.3.1 Surface Reconstruction Using Surface Suitability

To reconstruct the image surface segment S, ρ values are required. Given an arbitrary L, ρ values for surface s_i are computed by:

$$\rho(u) = \frac{I(u)}{\vec{n}(u)(\frac{L-p(u)}{\|L-p(u)\|})^t \imath}, u \in s_i \ . \tag{2.2}$$

The median of ρ values are used to obtain a single ρ value for image surface segment s_i. Then, the reconstructed image I_r for s_i is computed by:

$$I_r(u) = \rho_{s_i} min(\vec{n}(u)(\frac{L - p(u)}{\|L - p(u)\|})^t \imath, 0), u \in s_i \ , \tag{2.3}$$

we obtain the light source position L by minimizing the error E between the original intensity values I and the reconstructed intensity values I_r.

$$E_i = \sum_{u \in s_i} f(s_i) \|I(u) - I_r(u)\| \ \ , \tag{2.4}$$

$$E = \sum_{i \in S} E_i \ . \tag{2.5}$$

Unlike other methods [8, 129], where each image surface segment contributes equally to the total reconstruction error, we propose to assign weights f to each image surface segment based on its suitability to compute the LSP. The aim is to assign more importance to image surface segments which are more suitable for LSP estimation under the assumption of Lambert's law. To this end, surface attributes, characterize the surface suitability for LSP estimation, are extracted. Then, the weights f are learned using these attributes in a supervised learning. The surface attributes and learning procedure are detailed in Sections 2.3.2 and 2.3.3.

2.3.2 Surface Attributes for LSP Estimation

Lambert's law assumes a surface which diffusely reflects the light. The surfaces which satisfy this condition are more suited to estimate LSP. For instance, matte surfaces are preferred over glossy surfaces to estimate the LSP. The specular reflections are not considered by Lambert's law. Consequently the surfaces with highlight will negatively influence the LSP estimation. A cast shadow is caused by the occlusion of the light source position. That means that there are no light rays reaching the surface directly coming from the light source. Hence, intensity values would be misleading for LSP estimation (assuming Lambert's law). Therefore, LSP estimation from a shadow region would be prohibitive. Not only the photometric attributes, but also the geometric attributes of a surface is influential to estimate LSP. For instance, the surface normals on rough surfaces are prone to be more noisy than smooth surfaces. The intensity value is determined by the angle between the surface normal and the incident light direction. Therefore, noisy surface normals will negatively influence LSP estimation. Subsequently, smooth surfaces are preferred over rough surfaces.

It is clear that some surfaces have preferred attributes to estimate LSP. To this end, we define surface attributes. These attributes are further used in a learning scheme to measure the suitability

	Intensity	Chromatic	Normalized Chromatic	Hue
Representation	O_3	$[O_1, O_2]$	$[\frac{O_1}{O_3}, \frac{O_2}{O_3}]$	$\frac{O_1}{O_2}$
Invariant to	-	Highlights	Shadows	Highlights Shadows

Table 2.1: Opponent color space image representations and invariant properties [35].

of a surface to estimate LSP.

Photometric Representations

We aim to represent surfaces by their photometric attributes (e.g. glossiness). Moreover, it is important to identify surfaces under different photometric changes (e.g. highlights and shadows). The opponent color space is used to represent different photometric invariants [168, 51, 35]. The transformation of RGB to $O_1O_2O_3$ is given by:

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix}. \tag{2.6}$$

The different properties of the opponent color and their combinations are summarized in Table 5.1. The intensity information is represented by O_3. It has no invariant properties. Therefore, changes due to shadows and highlights are represented [168, 35]. Color information is contained in O_1 and O_2. Due to the subtraction in O_1 and O_2, they are invariant to shifts in illumination such as highlights [168, 35]. We consider invariance to illumination scaling (e.g. shadows) using normalized chromatic components ($\frac{O_1}{O_3}$ and $\frac{O_2}{O_3}$) as in [35]. Finally, we use $hue = \frac{O_1}{O_2}$ to account for both invariances.

Surface Attributes

Extracting material characteristics from images has been studied in [35, 29, 186, 153, 122]. These works consider material recognition as a texture classification problem. [99, 64] perform a detection in the image for common materials such as stone, wood, metal, fabric etc. Material characterization of a surface reveals important information about the surface property. For instance, in general, metal is hard and glossy whereas plastic is soft and matte. Surface attributes would highly benefit from material characterization. Therefore, in this chapter, material characterization is used to extract surface attributes to assign importance to the surfaces based on their suitability to estimate LSP. The opponent color and combinations (see Section 2.3.2), RGB and depth images are used to extract surface attributes (surface attributes related to [35] are extracted). We detail these attributes in this section.

Invariant Response. A cast shadow is caused by occlusion of the light source. Therefore, the light source position cannot be derived from a shadow region (assuming Lambert's law). Moreover, highlights are generated by specular reflections. The assumption of Lambert's law does not hold for highlight regions. To be able to distinguish image surface segments with shadows and highlights, we use the method proposed by [51] which measures the average gradient magnitude ratio defining the invariant response ($\frac{|\bigtriangledown O_3|}{\sqrt{|\bigtriangledown R|^2+|\bigtriangledown G|^2+|\bigtriangledown B|^2}}$ where $| \, . \, |$ stands for gradient magnitude). The image surface segments consist of uniform colors. Hence, the average gradient magnitude is expected to be low. However, shadows and highlights cause photometric edges. Therefore, the image surface segments with shadows and highlights are expected to have high invariant response.

Photometric Stability. Invariant representations contain instabilities. For instance, hue is unstable for colors with low saturation $\frac{O_1}{O_2}$ [35]. The surface reflection characteristics make the instabilities to vary for different surfaces. To account for the influence of instabilities, we use the method proposed by Everts et al. [35]. Mean intensity ($\mu(O_3)$) and saturation ($\mu(\sqrt{O_1^2 + O_2^2})$) statistics are considered to measure the photometric stability of a surface.

Interface Reflectance. Lambert's law assumes a surface which diffusely reflects the light. Interface (specular) reflectance is not defined by Lambert's law. Therefore, it is difficult to estimate the LSP from glossy surfaces. Moreover, the depth sensor is sensitive to glossy surfaces (e.g. shiny metal, mirror). The depth estimation becomes unstable on glossy surfaces. Therefore, surface normals are mostly noisy. To this end, we propose to extract an attribute which aims to detect glossiness of surfaces. Motoyoshi et al. [122] propose that the skewness (third-moment) of the intensity histogram is highly correlated with interface reflectance (gloss) and inversely correlated with diffuse reflectance (matte). Others, such as Sharan et al. [153] use the standard deviation whereas Dror et al. [29] use the kurtosis to account for interface reflectance. We also use skewness, standard deviation and kurtosis to measure the amount of interface reflectance using the O_3 component.

Colorfulness. Hue is invariant to shadows and highlights. Therefore, these photometrical changes should not influence the hue distribution of a surface segment. The assumption is that the albedo does not change within a surface segment (see Section 2.3.1 eq. 2.2). However, the variation in hue distribution most likely corresponds to the albedo change. Thus, the same albedo assumption may mislead the light source position estimation. To this end, we propose to use colorfulness by computing the hue entropy as in [35] ($-\sum(\mathbf{P}\log_2\mathbf{P})$). $\mathbf{P}$ represents the histogram of hue pixels.

Softness. Softness is useful to distinguish surfaces having diffuse (i.e. soft-plastic) or specular (i.e. hard-metal) reflection. Hu et al. [64] state that metal tends to have hard edges and sharp corners whereas plastic has soft edges and round corners. To this end, we measure the softness using the standard deviation of the gradient orientation ($\sigma(\bigtriangledown O_3)$) and magnitude ($\sigma(\mid \bigtriangledown O_3 \mid)$).

Texturedness. Most of the LSP algorithms use segmentation to group similar colored surfaces assuming that the pixels of the same surface segment have the same albedo. However, surfaces may also contain similarly colored textures such as crinkles in leather or grains in paper. These crinkles or grains will cause sharp intensity changes which may negatively effect the light source position estimation. To this end, we compute two Weibull parameters for the O_3 as proposed by Yanulevskaya and Geusebroek [186] to measure the amount of texturedness of a surface.

Micro-texture. The local non-uniformities on surfaces can be used to describe surface structure. Less micro-texture indicates polished glossy surfaces (e.g. metal) whereas more micro-texture indicates matte surfaces (e.g. fabric). Because of the diffuse reflection assumption of Lambert's law, these two types of surfaces are expected to influence the LSP estimation differently. The method proposed by Liu et al. [99] is used to measure the amount of micro-texture. In particular, we use the sum of residuals between a bilaterally smoothed O_3, $h(O_3)$, and the original O_3 ($\sum(h(O_3) - O_3)$).

Smoothness. Due to the imperfections of recording devices, the surface normals may be noisy on rough, crinkled and grained surfaces. Lambert's law uses the angle between the surface normals and light source position to minimize the error between re-rendered and original image. Noisy surface normals will negatively influence the error minimization. Smoothness aims at differentiating between rough and smooth surfaces. We use the statistics of the surface normals (from the depth image) to measure surface smoothness ($\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\vec{n_i} - \mu(\vec{n}))^2}$) ($N$ is the number of the points in a segment) and the mean of the gradient magnitude ($\mu(\mid \bigtriangledown\vec{n} \mid)$) of the surface normals. The first statistic is useful to observe overall deviation on an image surface segment whereas the second one is useful to observe local deviations. High values correspond to rougher surfaces. The surface normals are computed from the depth image using [147].

Area. The surface normals and intensity values may be noisy due to imperfections of recording devices. The variation of intensity distributions is important to alleviate these errors. Therefore, larger image segments are expected to contribute more to proper LSP estimation than smaller segments.

Attribute	Definition	Information Channel
Invariant Response	$\frac{\lvert\bigtriangledown O_3\rvert}{\sqrt{\lvert\bigtriangledown R\rvert^2+\lvert\bigtriangledown G\rvert^2+\lvert\bigtriangledown B\rvert^2}}$	$[O_3, RGB]$
Photometric Stability	$\mu(O_3)$, $\mu(\sqrt{O_1^2+O_2^2})$	$[O_1, O_2, O_3]$
Interface Reflectance	$Skew., \sigma, Kurt.$	O_3
Colorfulness	$-\sum(\mathbf{P}\log_2\mathbf{P})$	$\frac{O_1}{O_2}$
Softness	$\sigma(\bigtriangledown O_3)$, $\sigma(\mid\bigtriangledown O_3\mid)$	O_3
Texturedness	$[\gamma, \beta] = weibull$ [186]	O_3
Micro-texture	$\sum(h(O_3) - O_3)$	O_3
Smoothness	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\vec{n_i}-\mu(\vec{n}))^2}$, $\mu(\mid\bigtriangledown\vec{n}\mid)$	$depth$
Area	$\#pixels$	RGB
Surface Consistency	$L - \mu(L')$	$[O_3, depth]$
Curvedness	$\sigma(\mid\bigtriangledown\vec{n}\mid)$	$depth$

Table 2.2: Definitions of extracted attributes.

Surface Consistency. Estimated light source positions $L' = \{L_i\}_{i=1}^{m}$ (m is the number of the image surface segments in an image) should be consistent. Therefore, we express the surface consistency attributes by measuring the deviation from the average estimation for each image surface segment. For the i^{th} surface segment, surface consistency is measured by $L_i - \mu(L')$.

Curvedness. Curvedness distinguishes surfaces which have non-flat surfaces. Considering the richer surface normal representations of curved regions, we expect LSP estimation to be more precise for highly curved surfaces. To this end, using surface normals, the curvedness attribute is expressed by the standard deviation of the gradient magnitude ($\sigma(\mid\bigtriangledown\vec{n}\mid)$).

2.3.3 Image Surface Segment Suitability by Ranking

The suitability of an image surface segment is defined by the angular error between the estimated and the ground-truth light source position. The aim is to measure which image surface segment is preferred over others. Therefore, we consider the learning process as a ranking problem.

We use learning to rank (L2R) [101] to measure the suitability of different image surface segments for LSP estimation. The training set consists of image surface segments $S = \{s_i\}_{i=1}^{m}$ (m is the number of image surface segments) and ground-truth label y expressed in terms of angular error between ground-truth and estimated light source positions. Image surface segments are generated by using the (color-based) mean-shift algorithm. Attributes Φ are extracted from image surface segments as explained in Section 2.3.2. Φ and y are used to learn f which is described as follows:

$$f(s_i) = w^t \Phi(s_i). \tag{2.7}$$

The objective function to optimize the weight w is described as follows:

$$\min_{w} \frac{1}{2} w^t w + C \sum_{i=1}^{l} \xi(w; \Phi(s_i), y_i), \tag{2.8}$$

where ξ represents loss function. In this chapter, various loss functions are used and compared (see Section 2.5.1). The scores of f are used in eq. 2.4 to assign importance to the image surface segments. The image surface segments with high scores are more suitable for light source position estimation. Image surface segments which have a negative influence on LSP estimation (surfaces which have negative scores) are filtered out.

2.3.4 LSP Initialization and Search

The downhill simplex method [125] is used to minimize E in eq. 2.4. The state-of-the-art method [8] uses the camera view-point to initialize the light source position. However, to solve the minimization problem, a proper initial light source position is important to obtain fast convergence. Unlike the state-of-the-art [8], we propose to use constraints imposed by the $3D$ geometry to select the initial points. Assuming Lambert's law, the points on the object surface that receive the light close to a perpendicular angle have maximum intensity. Let O be the perpendicular projection of L on the planar surface. θ is the angle between the light source direction and the surface normal at surface point A. Then, $\cos(\theta) = I(O)/I(A)$, where $I(O)$ and $I(A)$ are intensity values at positions O and A. The distance d between points O and A is computed by the 3D coordinates, see Fig. 2.3. Finally, the height h of the light source is given by:

$$h = d \frac{\cos(\theta)}{\sqrt{1 - \cos(\theta)^2}}. \tag{2.9}$$

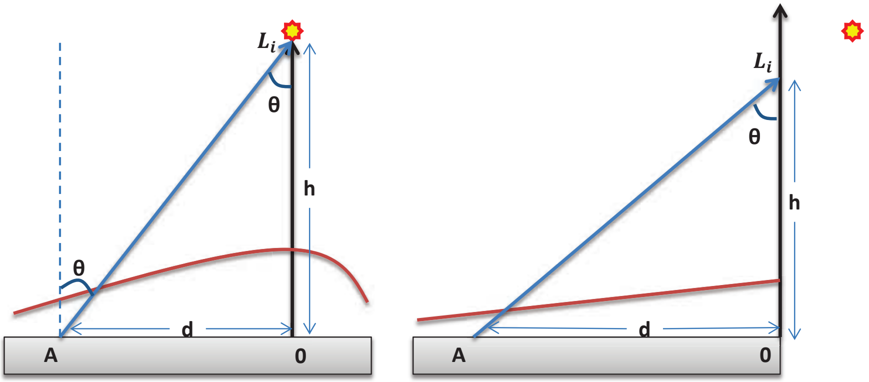


Figure 2.3: Initial light position L_i based on $3D$ geometry constraint. O is the point with maximum intensity on the surface. A is a random point on the surface of which the intensity is known. L_i is estimated as an initial guess for LSP which is on the direction of the surface normal of point O and satisfies Lambert's law for both points.

Initial light source positions are estimated for all the image surface segments in an image. These estimations are used to initialize the arbitrary light source position in eq. 2.3 by a weighted average. Moreover, to obtain surface consistency (see Section 2.3.2), it is necessary to have a LSP estimation for each image surface segment. $3D$ geometry-based initialization allows a speed-up of convergence for each individual estimation ($\approx 2\times$ faster).

2.4 LSP Estimation from RGB-D Sequences

Temporal information can be used to improve the accuracy of the single frame-based LSP estimation. We assume that L, with respect to static objects in the scene, does not change during a single video recording. Hence, the only change is the relative position of L with respect to the camera. Therefore, we propose to use the camera pose to provide temporal constraints in $RGB-D$ sequences. First, we estimate the camera pose to build correspondences between frames. Then, images are transformed to the same coordinate system to create consistency between estimations of different frames.

2.4.1 Camera Pose Estimation

Considering static objects in the scene, we propose to estimate the camera pose as a rigid body movement. In our framework, the iterative closest point (ICP) algorithm is used to estimate the camera pose [65]. ICP estimates the camera pose by aligning the data. Data alignment problem is treated as a nonlinear optimization problem in which correspondences between recordings (depth images) are approximated using the closest pairs of points found between successive depth images [130, 65]. After the computation of corresponding points, ICP aims to find a single transformation matrix $\mathbf{T}$ with minimal point-to-plane error [65]:

$$\arg\min \sum_{u} \|(\mathbf{T}v_i(u) - v^g_{i-1}(u))^t \vec{n}^g_{i-1}(u)\|^2. \tag{2.10}$$

The error is measured by how good each point $v_i(u)$ in the current frame fits the tangent plane at its corresponding point $v^g_{i-1}(u)$ in the previous frame [130, 65]. $\vec{n}^g_{i-1}(u)$ is the surface normal of the corresponding point $v^g_{i-1}(u)$ in the previous frame. Using a global coordinate g, the camera pose $\mathbf{T}$ is used to transform point $v_i(u)$ from the image coordinate to the global coordinate. Then, a linear approximation is adopted to solve this system. In our approach, a GPU-based implementation of ICP is used which provides real-time camera pose estimation.

2.4.2 Global LSP Refinement

After the camera pose is estimated by ICP, the proposed LSP estimation method is applied. To incorporate all video frames, L is transformed from local image coordinates to a global one. As a result, given an image I_i, its corresponding $\mathbf{T}$ and L, ρ values in s_j of eq. 2.2 are modified as follows:

$$\rho(u, \mathbf{T}) = \frac{I_i(u)}{\vec{n}(u)(\frac{\mathbf{T}L - p(u)}{\|\mathbf{T}L - p(u)\|})^t \imath}, u \in s_j. \tag{2.11}$$

Then, I_r, for a given s_j, is computed by:

$$I_r(u, \mathbf{T}) = \rho(u, \mathbf{T}) \min(\vec{n}(u)(\frac{\mathbf{T}L - p(u)}{\|\mathbf{T}L - p(u)\|})^t \imath, 0), u \in s_j, \tag{2.12}$$

$$E_{i,j}(u, \mathbf{T}) = \sum_{u \in s_j} f(s_j) \|I_i(u) - I_r(u, \mathbf{T})\|, \tag{2.13}$$

and the residual error of I_i is computed as follows:

$$E_i(u, \mathbf{T}) = \sum_{s_j \in I_i} E_{i,j}(u, \mathbf{T}). \tag{2.14}$$

Given an image sequences I_c, the energy function for the light source position is then defined by:

$$E = \sum_{I_i \in I_c} E_i(u, \mathbf{T}). \tag{2.15}$$

The estimated light source position is obtained by minimizing E given by eq. 2.15. Finally, the light source position which minimizes the residuals between the reconstructed and original video sequence is selected as the final estimation.

2.5 Experiments

Datasets and Evaluation Metric. The proposed light position estimation algorithm is evaluated on the dataset proposed by [8] and our newly collected dataset. Our dataset is collected using the Kinect camera resulting in 71 images. Three video sequences are recorded with known light source positions. Light source positions and objects are fixed for each video sequence but vary between different video sequences. A Philips daylight simulator bulb is used as the light source. As proposed in [8], the angular error between the estimated and ground-truth light source positions are used to measure the accuracy.

Implementation. Two types of learning to rank (L2R) methods are used, namely, pointwise and pairwise [101]. The main difference between these methods are their objective functions [78]. Pointwise methods aim to minimize the error based on single instances, whereas pairwise methods minimize the disorder between pairs. Pointwise methods require numerical scores for training labels, whereas pairwise methods use preferences between pairs. For pointwise and pairwise methods the $Liblinear$ [39] and Joachims [73] implementations are used respectively. The default parameter settings are used as provided by the implementations.

2.5.1 LSP Estimation from a Single RGB-D Frame

Experiment I : Influence of Learning Surface Attributes We evaluate our attribute-based LSP algorithm on the Boom13 dataset [8] and compare it with [8]. We follow the same steps for both algorithms. The main difference between the obtained results is that [8] gives equal weights to all image surface segments whereas our method assigns a weight to the each image surface segment based on its suitability to contribute to a correct LSP estimation. The results are summarized in Table 2.3. The results show that the proposed algorithm outperforms [8]. The significant improvement over [8] indicates the importance of surface attributes to estimate LSP. Hence, image surface segments influence LSP estimation differently based on their appropriateness.

Experiment II: Influence of Learning Algorithms In this experiment, we evaluate three different learning algorithms to rank the image surface segments. Support vector classifier SVC and

Method	Performance (Mean Angular Error)
Boom et al. [8]	12.9°±10.6°
Proposed-SVC10	9.4°±5.9°
Proposed-SVC15	9.2°±5.9°
Proposed-SVC20	**8.6**°±**5.5**°
Proposed-SVC25	8.7°±5.2°
Proposed-SVR	9.4°±6.7°
Proposed-RankSVM	8.8°±**5.1**°

Table 2.3: LSP estimation performance on *Boom13 dataset*. There is no threshold for angular error to consider an image surface segment to be good/bad. Varying thresholds are used to specify positive or negative labels. The numbers next to SVC represent the angular error threshold used. The proposed attribute-based LSP algorithm outperforms [8] which assumes equal importance to all surfaces. Various learning algorithms are also tested, namely, support vector classifier SVC, support vector regressor SVR and $RankSVM$. The results show that learning the surface attributes outperforms the method without learning [8] regardless the choice of the learning algorithm. $SVC20$ performs best.

support vector regressor SVR are used as pointwise methods. $RankSVM$ [73] is used as a pairwise method. These algorithms differ mainly by their loss functions $\xi(w; \Phi(s_i), y_i)$ of eq. 6.5. ξ for SVC, SVR and $RankSVM$ are $\max(0, 1 - y_i w^t \Phi(s_i))$, $(\max(0, |y_i - w^t \Phi(s_i)| - \epsilon))^2$ and $\max(0, 1 + w^t \Phi(s_i) - w^t \Phi(s_j))$ respectively [78]. The sensitiveness of the loss functions are determined by ϵ parameters. $\Phi(s)$, w and y stand for the surface attribute, the weights and the labels respectively.

The angular error between the estimated and ground-truth light source positions are used as training labels. Since there is no threshold for the angular error to determine an image surface segment to be good/bad, we use varying thresholds for the angular error to specify image surface segments to be positive or negative labels for SVC. The angular errors are directly used as training labels for SVR. $RankSVM$ requires pairwise preferences between image surface segments. These preferences are created based on their angular errors.

The results show that learning the surface attributes outperforms the method without learning [8] regardless the choice of the learning algorithm (See Table 2.3). This indicates the importance of learning surface relevance for LSP estimation. SVC using 20° error threshold performs best. However, the necessity of choosing a labeling threshold makes SVC less practical. $RankSVM$ performs as good as SVC without introducing any hand-crafted rules for labeling. Therefore, $RankSVM$ is used for the rest of Chapter 2.

Experiment III: Influence of Surface Attributes In this experiment, we study the influence of

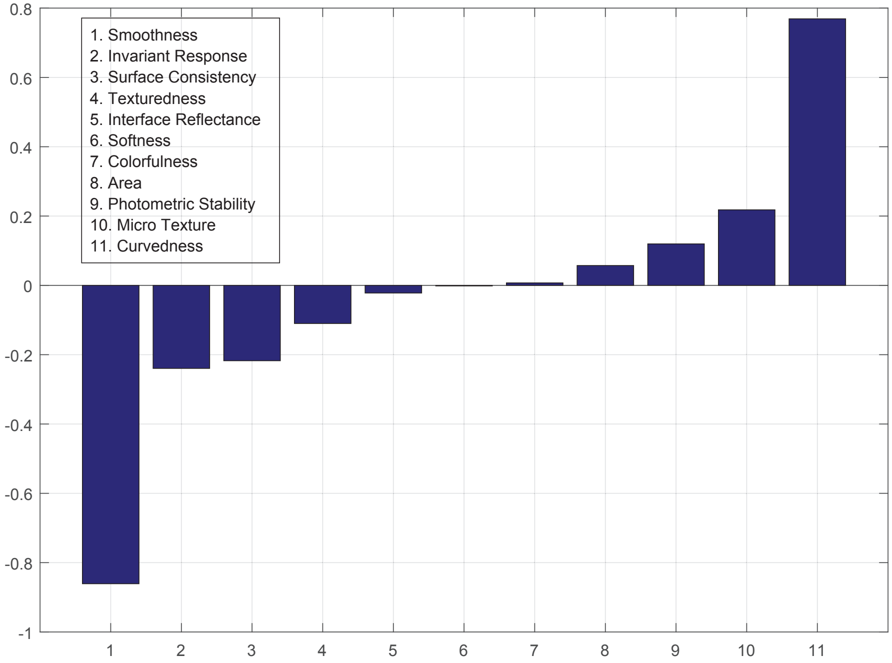


Figure 2.4: Attribute weights: The weights are obtained by averaging the summed classifier weights of different dimensions of the same attributes. It illustrates that surface attributes influence the LSP estimation differently.

each individual surface attributes. The weights are obtained by averaging the summed classifier weights of different dimensions of the same attributes. The attribute importance is summarized in Fig. 2.4.

Deviation from smoothness of a surface is observed to be negatively related to the correctness of the estimation. This is due to noisy surface normals extracted from rough surfaces mislead the optimization algorithm. Moreover, rough surfaces are more prone to cast shadows due to the occlusion of the light source.

The surfaces are segmented based on their color. Shadows and highlights cause gradient changes within the same colored segments. Invariant response takes into account this. Higher average gradient ratio corresponds to shadows and highlights. Considering Lambert's law, it is expected that light source position estimation is negatively affected by invariant response.

Surface consistency is defined by the deviation of an image surface segment LSP estimation from the average estimation of the other image surface segments in the image. The results show the importance of a global consistency condition. Another conclusion is that proper image surface segments vote for similar light source positions. Deviating from the average estimation of image surface segments negatively influence the importance of an image surface segment.

The amount of texturedness on LSP estimation is important. This is due to the surface homogeneity assumption of LSP algorithms. They assume that the intensity changes are caused by shading. However, changes caused by the surface texture negatively influence the optimization algorithm to reach convergence. Therefore, as expected, less textured regions are more useful for LSP estimation.

Interface reflectance is highly correlated with surface property of being matte or glossy. Surfaces become more glossy with an increasing amount of interface reflectance. Since LambertŠs law assumes a surface which diffusely reflects the light, the light position estimation is negatively affected by interface reflectance.

An increase in the softness-hardness scale actually indicates an increase of the hardness of the surface. Therefore, it can be concluded that the material hardness negatively affect the LSP estimation. This can be explained by the interrelation of hardness of a material with other properties. For instance, a metal is of harder material than plastic and it is more likely to generate a more glossy surface than plastic.

LSP algorithms benefit from increasing the size of an image surface segment. This is because increasing the area size also increases the variation in intensity distribution which alleviates the errors due to the imperfections of recording devices.

The surfaces being photometrically stable has a positive influence on the light source position estimation.

Less micro-texture mostly indicates polished, glossy surfaces (e.g. metal) whereas more micro-texture indicates matte surfaces (e.g. fabric). Because glossy surfaces are more prone to be affected by interface reflectance, the amount of micro-texture positively influences the LSP estimation.

The amount of curvedness has a positive affect on the LSP estimation accuracy. Curved surfaces create more variations of surface normals. This provides intensity variations even for small regions. Whereas a flat surface usually changes monotonically and does not create such intensity variation.

2.5.2 LSP Estimation from a RGB-D Video Sequence

Experiment I: Influence of Temporal Constraints In this experiment, we conduct two experiments. First, we compare the performance of the proposed attribute-based LSP estimation algorithm with [8]. Second, we compare the performance of LSP estimation based on a single frame with video sequence. The results are summarized in Table 2.4: The results for the first experiment show that our attribute-based method (mean angular error 7.1°) outperforms [8] (mean angular error 12.9°). For the second experiment, we estimate a single global light source

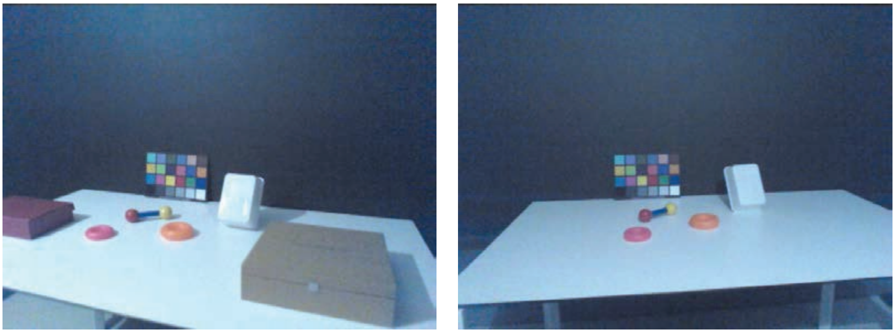


Figure 2.5: Sample images from our $RGB-D$ video dataset. Images are from different video sequences.

Method	Performance (Mean Angular Error)
Boom et al. [8]	12.9°±6.0°
Proposed Attribute	7.1°±2.7°
Boom et al. [8] + Proposed Temporal	9.7°±5.7°
Proposed Attribute+Temporal	**6.0°±2.5°**

Table 2.4: LSP estimation performance on *Our dataset*. The proposed attribute-based method outperforms [8]. The proposed temporal constraints improves the accuracy of the proposed attribute-based method and an off-the-shelf LSP estimation method [8].

position for the whole video sequence using the proposed temporal constraints. To obtain the light source position for a single image, the estimated global light source position is transformed into local image coordinates (using the estimated camera pose). Then, the errors are measured. For each sequence, our temporally constrained LSP algorithm reduces the LSP estimation error from 7.1° to 6.0°.

Experiment II: Improving off-the-shelf LSP Estimation Method In this experiment, we use a state-of-the-art LSP algorithm [8] and apply the proposed global refinement step. The objective function is replaced by the proposed temporal constraint. The mean error is reduced from 12.9° to 9.7° with respect to the original LSP algorithm [8] (See Table 2.4).

2.5.3 Synthetic Object Rendering

Rendering an object into a $2D$ image without knowing the light source is a difficult task. The estimated light source position allows us to realistically render synthetic objects into the scene. In particular, we can correctly incorporate the photometric effects (e.g. shadows and shadings).

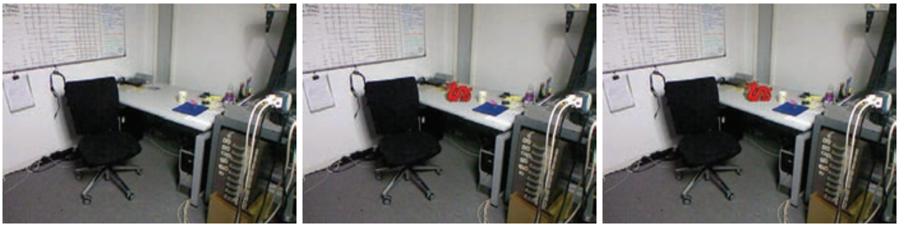


Figure 2.6: An illustration for rendering a synthetic object into the scene. The original image is on the left. The ground-truth light source position is used to render object and shadows into the scene in the middle image. Estimated light source position is used in the right image.

We use the estimated light source position to generate shading on a synthetic dragon object and shadow generated due to scene geometry. Fig. 2.6 illustrates that the rendered object has realistic appearance.

2.6 Conclusion

In this chapter, we have exploited the influence of surface attributes on the accuracy of LSP estimation. Given a single $RGB - D$ image, we first analyzed the effects of photometric and geometric surface attributes. Then, surfaces are ranked using a supervised learning scheme. The ranking results are used to decide the contribution of an image surface segment for LSP estimation. Higher importance is assigned to those image surface segments which have suitable photometric (i.e. Lambertian reflectance) and geometric surface attributes. To speed up the LSP estimation, a geometry constrain has been introduced to initialize point selection. Moreover, the image surface segments which have a negative influence on LSP estimation are filtered out. Additionally, we introduce a temporal constraint to estimate LSP from a $RGB - D$ video sequence. LSP is optimized using the camera poses between successive frames. The results show that our method based on weighting image surface segments using their attributes outperforms the state-of-the-art methods. By using the proposed surface weighting, the angular error is reduced from $12.9°$ to $8.6°$ and $12.9°$ to $7.1°$ for *Boom* and our newly collected datasets respectively. Moreover, using the camera pose to temporally constrain LSP reduces the angular error $(6.0°)$ compared to using single frames $(7.1°)$.

3

Per-patch Metric Learning*

3.1 Introduction

Viewing and lighting condition changes in real-world scenes cause substantial variations in image feature representations. Significant progress has been made in developing image representations that are invariant to transformations such as photometry [48, 49] or geometry [7, 98, 115, 194]. Image representations invariant to such changes are beneficial for applications such as object recognition, image retrieval and scene recognition.

A full invariant representation, unfortunately, leads to a decrease in discriminative power [172]. This drawback is due to distinguishing transformations that a full invariant representation cannot capture. For example, under rotational invariance a "6" is identical to a "9", and under shading invariance the texture of "grass" turns into "moss". Another disadvantage of invariant image representations is that they negatively influence stability [50, 170, 36]. This is due to their sensitivity to noise when the image signal is low or ambiguous. For example, a rotational invariant based on the dominant orientation [115] becomes unstable when multiple equally dominant orientations are present. Illumination invariant representations based on intensity normalization such as normalized-rgb or hue [49] become unstable for low intensity values.

Current invariant methods are always "on". One can either choose to use the invariance, or choose not to use it. There is no middle-ground. It is not possible to have invariance for only some shading or only slight rotations. These rigid properties of current invariants play a central role in the trade-off between invariance and discriminative power. Here, we propose to replace these binary on/off invariants, by steering the invariance to a limited range of disturbances. Such a limited degree of invariance is called *robustness*. For example, in the case of rotation, the proposed method can determine that a "6" is only invariant up to $\pm 45^o$ of (and thus robust to)

[*] Published as S. Karaoglu, I. Everts, J. C. van Gemert, and T. Gevers, "Per-patch Metric Learning for Robust Image Matching," In the International Conference on Image Processing (ICIP), Quebec, 2015.

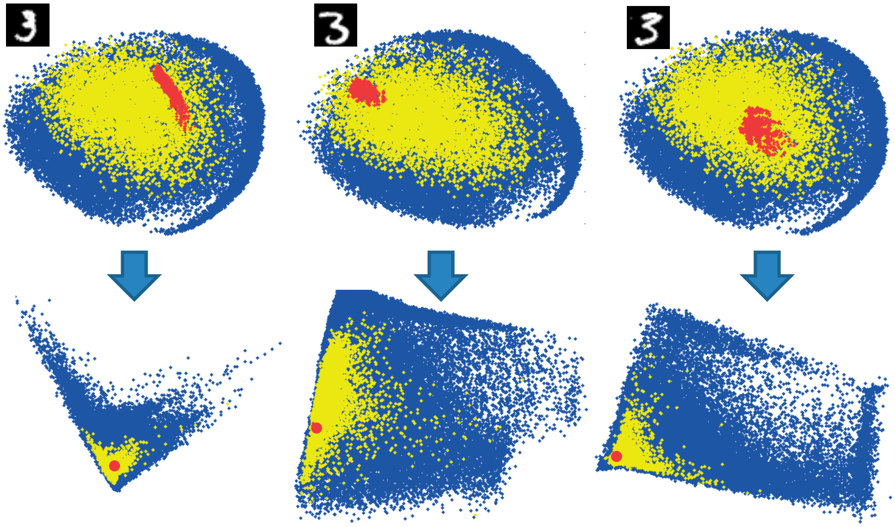


Figure 3.1: 2D PCA projection of SIFT extracted from dataset samples (blue); 1000 affine transformations of the top-left image patch (red); same-class samples (yellow). The top row is the original space, the bottom row is after learning the metric.

rotation, therefore eliminating the confusion with "9".

By allowing a degree of invariance, a single global image representation cannot be used as it depends on the specific image content how the limited transformation range will take effect, which is illustrated by the "6" and "9" example. Therefore, the proposed method is required to achieve robustness on a per-patch basis. Fig. 3.1 (top) illustrates that feature distributions after a transformation depend on the patch content, since even instances within the same class behave differently (red versus yellow). The bottom row of Fig. 3.1 illustrates the effect of steerable invariance applied to each patch.

To achieve robustness, we compute a Mahalanobis metric for each individual patch. In effect, the metric weights the subset of feature dimensions that require robustness. For this, a relevant subset of transformations is generated and a metric that is specific for only those transformations is learned. We present two approaches for learning the metric: (i) *full* and (ii) *direct*. The *full* method generates synthetic image patches, extracts descriptors for each patch, and obtains robustness through a metric that is learned on these descriptors. In the *direct* approach, we generate a transformation map only once, and use this map to directly estimate the metric from the patch without explicitly generating any synthetic images.

3.2 Related Work

Approaches that aim to achieve (full) invariance either use a transformation model based on the laws of physics [48, 49] or a model of the observed variations [7, 98, 102, 115]. The two disadvantages of invariants, stability and discriminative power, can be addressed by propagating camera noise parameters [50] or by deriving quasi-invariants [170]. Noise propagation requires proper noise estimation and the quasi-invariants are incomparable over different images and thus cannot be used for matching.

In contrast to employing pre-determined models, (deep) learning methods learn invariant features from unsupervised training examples [85, 140, 149]. Such methods do not explicitly model invariance as they attain robustness from training examples. Therefore, learning methods require large amounts of training data which is hard to obtain. Moreover, learning approaches do not directly incorporate known physical laws of the world. In this chapter, we use a hybrid approach of modeling robustness by learning from synthetically generated geometric and photometric data.

Synthetically generated data can be used to directly create variation in the train and test samples [11, 45, 93]. Other brute-force methods like ASIFT [121] generate a full range of affine transformations for both training and testing images which are used in an exhaustive matching scheme. Similar to our work, Simard et al. [157] avoid brute-force approaches and use synthetically generated images to learn a robust distance metric which is tangent to the manifold that is spanned by the generated transformations. We also learn a robust metric, however, where Simard et al. [157] require pixel values to estimate a manifold, our method estimates a Mahalanobis distance, which is applicable to any feature representation such as SIFT. To improve discriminability of a local descriptor, Cai et al [11] also propose to learn a projection matrix for a limited range of affine transformations through generated data. However, it is important to note that the authors learn a global projection whereas Chapter 3 proposes to learn patch specific projections. Fig. 3.1 illustrates that the same transformations applied to even the same class instances has different effects for different patches. These variations are thus patch-dependent and might not reflect the appropriate effect on other patches from the same class.

Chapter 3 has following contributions: (i) we demonstrate that a full invariant representation leads to a decrease in the discriminative power of a descriptor. Accordingly, we propose to limit the degree of invariance and augment the discriminativeness of a descriptor. (ii) we demonstrate that a single global image representation cannot be used as the effect of transformation essentially depends on the specific image content. Thus, a patch specific metric is proposed. (iii) we propose two alternatives to learn per-patch metric: by either explicitly applying transformations or obtaining directly from the patch.

3.3 Per-Patch Metric Learning

We first develop a metric that is learned from synthetically generated transformations of photometric and geometric distortions.

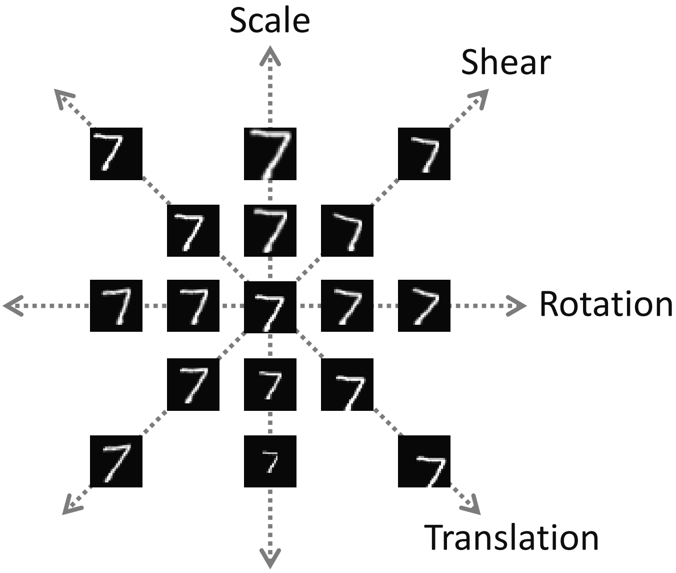


Figure 3.2: Transformations used to steer geometric invariance. The patch in the center is the original image, whereas the others are geometrically transformed versions.

Geometric Transformations. Images are subject to geometric distortions introduced by perspective effects caused by view point changes. For small patches, the perspective transformation $(x', y')^T$ can be approximated by an affine transformation for a given point $(x, y)^T$ as

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x \cos\alpha & -\tau_x \sin\alpha & t_x \\ \tau_y \sin\alpha & s_y \cos\alpha & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \tag{3.1}$$

where s denotes scale, τ represents shearing, α is the rotation angle, and t denotes translation. Examples of the transformations are depicted in Fig. 3.2. Sample generation involves repetitive random selection from appropriate parameter ranges.

Photometric Transformations. To model photometric changes, we assume Lambertian reflection. Accordingly, the color response (I) for the visible spectrum (λ), using a camera with spectral sensitivity (f) and an illumination source with the spectral power distribution (e) can be defined as

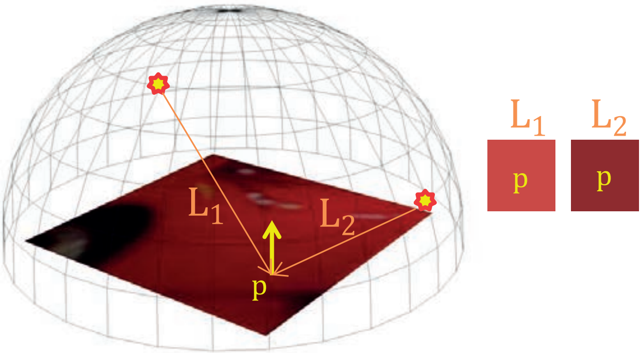


Figure 3.3: Illustration of photometric changes. The color response of point p changes due the angle between the incident light (L) and surface normal (yellow arrow) at the point p. According to Lambert's law, if this angle becomes smaller, the color response becomes brighter.

$$I = \vec{n}.\vec{s} \int_{\lambda} e(\lambda)\rho(\lambda)f(\lambda)d\lambda. \tag{3.2}$$

s, n and ρ denote the illumination direction, the surface normal and the surface albedo respectively.

To obtain photometric robustness, we generate variations caused by Lambert's Law. For the same surface patch the viewpoint and illuminant spectral power distribution are the same. The color response can only vary due to changes in illumination direction (i.e. ρ, f, e and n remain constant). Therefore, the changes in I can be modeled by illuminating the patch from different positions (See Fig. 3.3).

The center of the image patches are considered to be at $(0, 0, 0)$ and placed perpendicular to the light source position. Then, the light source position is systematically sampled within a certain radius. The patches are sufficiently small to be assumed planar. Hence, surface normals are equal for the patch under consideration. Thus, the light source direction is the only factor determining the effect of the photometrical changes.

3.3.1 Metric Learning

A Mahalanobis distance metric between image features $\mathbf{x}_i$ and $\mathbf{x}_j$ is parameterized by the matrix M,

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T M(\mathbf{x}_i - \mathbf{x}_j)}, \tag{3.3}$$

where M is a positive semi-definite matrix. A straightforward approach to compute M is to use $M = C^{-1}$ where C is the empirical feature covariance of the training data [6, 88]. The rationale behind using the inverse covariance as a metric is that a high variance in a feature dimension means that this dimension is not very stable. The most informative dimensions are those that have low variances.

For large-scale matching it is convenient to use fast indexing techniques such as trees [114]. Such techniques typically work with the Euclidean distance. To this end, the metric can be rewritten to

$$(\mathbf{x}_i - \mathbf{x}_j)^T W^T W (\mathbf{x}_i - \mathbf{x}_j) = ||W\mathbf{x}_i - W\mathbf{x}_j||^2, \tag{3.4}$$

where $W = M^{-\frac{1}{2}}$. This effectively scales the feature space with an affine transformation W to allow the Euclidean distance to be used for metric M.

Robustness is obtained by generating a limited range of transformations for a single patch. From these generated samples C is estimated, which is then used to compute M.

We coin this simple method for metric learning the *full* approach, since it needs to fully generate a large number of transformations for a patch in order to extract the image features and estimate the covariance matrix.

3.3.2 Direct Metric Estimation

Instead of using the brute-force approach of computing C by explicitly generating patches and extracting features from them, we propose a *direct* approach to estimate the metric per-patch. In the direct approach, the covariance is estimated from a single patch.

Let $\mathbf{x}$ be an image feature vector as a column, with N dimensions. For clarity, we start with pixel values to explain the direct metric estimation, i.e. $\mathbf{x}$ is a vector of pixels. Later we show our method readily applies to other descriptors.

Pixel Values. The direct metric is a combination of two terms, a transformation probability D_T that a pixel moves to a different position after transformation T, and the patch-specific covariance term V of the feature values.

Let D_T be a symmetric matrix of size NxN, containing the probabilities $P_T(i|j)$ of a pixel at position j affecting the position of pixel i under a transformation T. Note that this transformation is independent of the actual feature values. Matrix D_T represents the per-pixel transformation probability which is determined by simulating a large set of transformations and comparing the transformed patch with the ground truth location obtained through the homography in eq 3.1.

The matrix V of size NxN is the variance matrix with elements $\sigma(i, j)$ representing the covariance of pixel values at position i with respect to position j. To compute the covariance matrix, we need the expected average weighted pixel value $\mathbf{x_i}$ at position i after the transformation T, which is given by:

$$\mathrm{E}[\mathbf{x}_T(i)] = \sum_{j=0}^{N^2} P_T(i|j)\mathbf{x}(j) = D_T\mathbf{x}^T, \tag{3.5}$$

where $\mathbf{x}_T$ represents the pixel vector $\mathbf{x}$ after the transformation T. The expected value of the transformed image $\mathbf{x}_T$ is denoted by $\mathrm{E}[\mathbf{x}_T(i)]$ and represents the average image of all transformations that are present in D_T. The covariance $\sigma(i,j)$ after transformation T is then

$$\sigma(i,j) = \mathrm{E}\big[\big(\mathrm{E}[\mathbf{x}_T(i)] - \mathbf{x}_T(i)\big)\big(\mathrm{E}[\mathbf{x}_T(j)] - \mathbf{x}_T(j)\big)\big]. \tag{3.6}$$

Note that in the transformation of $\mathbf{x}_T(i)$ and $\mathbf{x}_T(j)$ it is allowed for pixels to move independently to other pixels. Eq 3.6 can be rewritten in matrix form to obtain V directly

$$W_V = [[D_T > 0]] \bullet (D_T\mathbf{x}\mathbf{1}^T - (\mathbf{x}^T\mathbf{1})^T), \tag{3.7}$$

$$V = \frac{1}{N^2} D_T \bullet W_V W_V^T, \tag{3.8}$$

where $\mathbf{1}$ is a column vector of all ones, $\bullet$ denotes element-wise multiplication and $[[\cdot]]$ indicate Iverson brackets which resolves a (matrix) element to 1 when the argument is true, and 0 otherwise. The metric is computed by $M = V^{-1}$, and the transformation by $W = V^{-\frac{1}{2}}$.

SIFT. For other descriptors, the D_T matrix of transformation probabilities can be reused and is not required to be recomputed. The V matrix, however, has to be adapted to the specific form of the descriptor. In the case of SIFT, D_T is converted to D_T^{sift} of size 128x128. In contrast to generating all possible sift values, D_T^{sift} has to be computed once only.

The 128 dimensions of SIFT comprise of a 4x4 spatial grid and 8 angular bins (4x4x8). We use the pixel-based transformation probabilities D_T to directly calculate the probability $P_T^{\text{sift}}(i|j)$ for spatial SIFT bins i and j with

$$P_T^{\text{SIFT}}(i|j) = \sum_{y=0}^{N^2} [[f(x) = i]][[f(y) = j]]P_T(x|y), \tag{3.9}$$

where the function $f(x)$ maps the pixel at location x to the correct SIFT-bin i. For the angular transformation probabilities, we assume independence with the spatial bins. The unit circle is sampled with 360 vectors and transformations are applied to these vectors. Since the original orientation is known, this results in counting how often a vector switches bins after the transformations. The joint 128x128 matrix D_T^{sift} is calculated by multiplying the angular probabilities with the spatial probabilities.

3.4 Experiments

Dataset and Implementation Details. The *full* and *direct* metric methods are evaluated on the ALOI dataset [47] for SIFT descriptor robustness against geometric and photometric distortions. The ALOI dataset contains 1000 objects under varying imaging conditions. We use variations due to camera viewpoint and illumination direction as geometric and photometric distortions respectively. To annotate matching pairs, the same procedure is followed as in [35]. In total 8300 matching pairs are extracted of which 200 are used for validation. Classification is performed by feature matching in a 1-NN classification scheme.

3.4.1 Geometric Robustness

First, we evaluate per-patch metric learning under geometric distortions. The performance of the proposed methods is compared against the original SIFT and other methods.

The optimal parameter ranges are obtained on the validation set. Parameters are repetitively sampled (1500 times) from various ranges to either generate and apply geometric transformations in the *full* approach or to estimate the transformation probabilities $P_T(i|j)$ in the *direct* approach. A joint optimization of parameters on the validation set yields a translation in [-2:2] and shearing in [-.1:.1] for the SIFT descriptors. Scale and orientation do not affect the performance as the positional difference between viewpoints does not yield large scale and orientation variations.

The results in Table 3.1 show that the proposed *full* and *direct* methods have a significant improvement of 6.57% and 6.22% over the SIFT performance respectively. The significance is validated by t-test a ($p < 0.001$). The substantial performance increase is due to the fact that viewpoint variations degrade the SIFT performance for matching. Considering full-rotation invariance for SIFT leads to a dramatic performance loss as the most discriminative information is ignored and due to the visual complexity it is harder to estimate a dominant gradient orientation.

Additionally, we evaluate raw pixels and tangent distance (TD) [157] on this set. We obtain a classification rate of 16.54% and 23.47% respectively. As discussed in Section 3.2, Simard et al. invoke prior information by generating small global transformations. The performance improvement over raw pixel values supports our idea of exploiting prior information for steering the invariance. However, as expected, the raw pixel matching performance is far beyond the SIFT performance which makes TD less applicable when it is necessary to use features except raw pixel values.

3.4.2 Photometric Robustness

We evaluate the proposed *full* metric for photometric robustness. The *full* method operates by generating synthetic samples as explained in *Photometric transformations*. Matching patch pairs are selected from the same viewpoint but with different illumination. The same evaluation pro-

SIFT	+Rot. Inv.	Proposed *full*	Proposed *direct*
54.85%	13.86%	**61.42%**	61.07%

Table 3.1: Matching performance on geometric distortions. Left to right: original SIFT descriptor, full-rotation invariant SIFT descriptor, proposed *full*(synthetic) and *direct* methods. Note that full-rotation invariance is unstable whereas limited range of invariance is stable.

cedure is followed as in Section 3.4.1. The results are shown in Table 3.2. The proposed *full* method significantly outperforms the SIFT performance (t-test with $p < 0.001$).

SIFT	Proposed *full*
73.4%	**77.86%**

Table 3.2: Matching performance on photometric distortions. "full" outperforms the SIFT performance.

3.5 Conclusion

In Chapter 3, we propose a generic patch-specific robust metric learning method to improve matching performance of local descriptors. We show that a full invariant representation leads to a decrease in discriminative power of descriptors. Therefore, we propose a per-patch metric learning method that is invariant to only a range of variations. We propose to learn a patch specific a Mahalanobis metric. Two approaches for learning the metric are presented: (i) *full* and (ii) *direct*. The proposed approaches are validated on ALOI dataset for two different image transformations, namely, geometric and photometric. It has been shown that the proposed approaches outperform the original SIFT descriptor matching performance.

4

Text Detection for Fine-grained Classification and Logo Retrieval*

4.1 Introduction

Many existing object recognition methods are focused on distinguishing definite objects such as horses, bicycles and cars [34]. Object recognition results obtained for different benchmarks, e.g., Pascal VOC, show that there has been a significant progress to recognize these “distinct” object categories. However, the performance of these methods may deteriorate to distinguish categories of objects that only slightly differ in appearance, such tasks include fine-grained classification. Fine-grained classification is the problem of assigning images to sub-ordinate classes in which objects differ only in (subtle) details (e.g. flower types [132], bird species [187]). Although, visual cues (e.g. color, texture and shape) can be used to distinguish visually distinct objects, the same visual cues may lack discriminative power to differentiate object types of similar appearance.

In Chapter 4, we address the problem of fine-grained object classification by combining textual and visual cues. In particular, we focus on the classification of *Buildings* into their sub-classes such as *Cafe*, *Tavern*, *Diner*, etc. The reason to use textual cues for such task is that text adds semantics beyond visual cues. For instance, in Fig. 4.1, the aim is to classify the three images based on their semantics. In this case, visual cues are not sufficient or even misleading as the first two images have similar scene appearances. Textual cues are useful to recognize that the two

*S. Karaoglu, R. Tao, J. C. van Gemert and T. Gevers, "Con-Text: Text Detection for Fine-grained Classification and Logo Retrieval," Under revision in IEEE Transactions on Image Processing (TIP), 2016. Ideas previously appeared in: S. Karaoglu, J. C. van Gemert, and T. Gevers,"Object Reading: Text Recognition for Object Recognition,"In the European Conference on Computer Vision Workshop (ECCV-W), Firenze, 2012. S. Karaoglu, J. C. van Gemert, and T. Gevers, "Con-Text: Text Detection Using Background Connectivity for Fine-Grained Object Classification," In ACM International Conference on Multimedia (ACM MM), 2013.

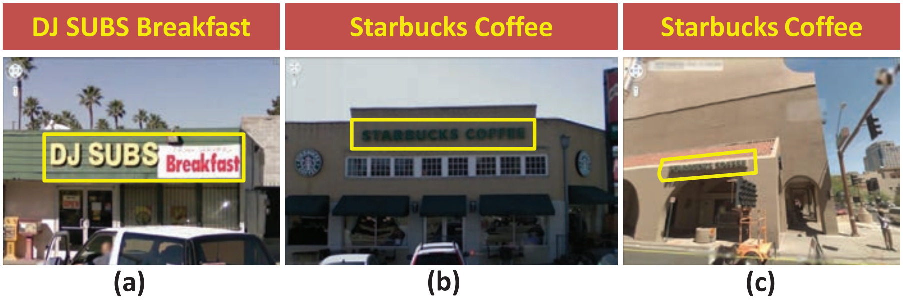


Figure 4.1: An example of fine-grained *Building* classification [83]. Visual cues would group (a)-(b) whereas scene text reveals the semantics and clusters (b)-(c).

(right) images belong to the same category since they contain the same brand name *Starbucks*. Therefore, we propose to use both textual and visual cues.

The success of the proposed fine-grained object classification method (fusion of visual and text modalities) highly depends on the correctness of the extracted textual image cues. Therefore, a robust character localization is required. The state-of-the-art text detection methods [178, 96, 31, 128, 87, 191, 179, 69, 103] extract geometric, structural and appearance properties from candidate text regions which are obtained using a connected component or sliding window approach. These regions are further verified using the extracted features if they contain text or not. In contrast, we focus on the scene background (non-text regions) rather than text regions. Our motivation is that the majority of the scenes consist of background pixels e.g. 93% of ICDAR13. Moreover, background pixels are mostly homogenous within themselves e.g. fences, boards, roads, buildings, windows etc. and highly contrasted with text regions. Focusing on eliminating background pixels using background connectivity rather than directly detecting text regions benefits from eliminating larger number of non-text regions at a reduced risk of eliminating true text regions (due to contrast). Moreover, since the proposed method does not extract text specific features, it does not require any tuning for varying text size, style and orientation.

Text is designed to attract human attention. This has been verified by Judd et al. [76], and Wang and Pomplun [177] who show that text features are more discriminative than generic object features and scene text receives human attention more than other generic objects. Accordingly, in this chapter, we consider salient and non-salient regions as text and background regions respectively. The proposed method initially selects background (non-text regions) seeds using color and curvature saliency, and spatial context information. Then, it detects the background from these seeds based on background connectivity. Once the background has been detected and eliminated, text regions are identified. The detected character candidates are further processed by two methods to perform text recognition i.e. ABBYY commercial OCR machine and state-of-the-art character recognition algorithm [69]. Then, spatial pairwise constraints between character candidates are used to obtain textual cue representations. Finally, extracted textual cues are used in combination with visual cues for fine-grained classification and logo retrieval. The pipeline of the proposed method is summarized in Fig. 6.1.

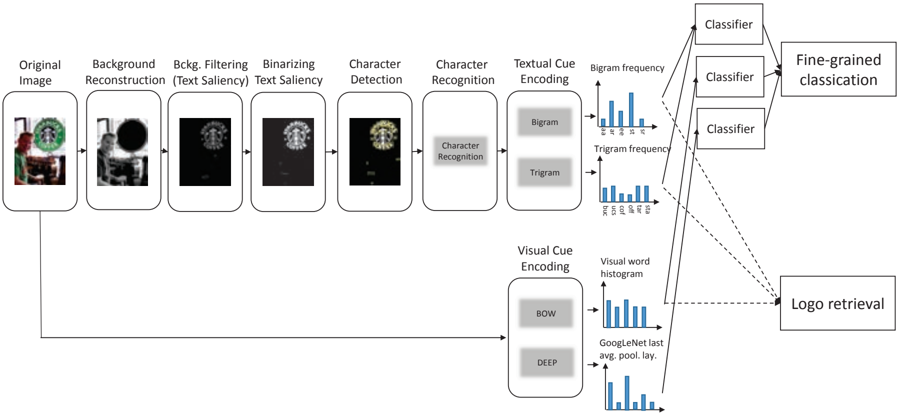


Figure 4.2: The flow of the proposed method. We propose a generic, efficient and fully unsupervised text detection algorithm by eliminating scene background. Detected text regions are processed by a state-of-the-art character recognition algorithm. Then, bi- and trigrams (textual cues) are formed between the recognized characters by considering spatial pairwise constraints. Extracted textual and visual cues are combined for fine-grained classification and logo retrieval.

Chapter 4 has seven main contributions:

- We propose a generic and computationally efficient character detection algorithm without any training involved. Unlike the state-of-the-art text detection methods which try to detect scene text directly, the proposed method detects the background to infer the location of text.

- We experimentally show that removing scene background reduces clutter and subsequently improves the character recognition performance of standard OCR systems. Moreover, removing the background reduces the search space allowing the extraction of computationally expensive features for character recognition.

- We propose a fine-grained classification approach which combines textual and visual cues to distinguish objects. To the best of our knowledge, this is the first approach to combine textual and visual cues of objects in images.

- We are the first to combine textual and visual cues for logo retrieval in natural scene images.

- We propose to constrain textual cues by spatial information. We show that encoding textual cues with proposed constraints is superior than without these constraints.

- We introduce a new challenge (fine-grained place of business classification) and a dataset. The introduced dataset, extracted features (textual and visual) and text detection code are publicly available[†].

[†] https://staff.fnwi.uva.nl/s.karaoglu/datasetWeb/Dataset.html

- We provide a large text detection dataset (10K images with 27601 word boxes).

Preliminary versions of this chapter appeared in [82, 83]. We extended our earlier studies in different aspects: 1) In [82, 83], there are no spatial constraints to form textual cues. Characters, ’P’, ’E’, ’A’, ’T’, can be combined into ’TAPE’, ’PATE’, or ’PEAT’. It is not possible to distinguish between ’TAPE’, ’PATE’ and ’PEAT’ by considering all possible bigram combinations. Hence, we propose to encode textual cues using spatial constraints. 2) Location information of the recognized characters are essential to perform spatially constrained encoding. However, ABBYY is used as a character recognition system in [82, 83]. Character location information is not provided by ABBYY. This restricts to perform proposed encoding. Therefore, a different character recognition algorithm [69] is used. A method is proposed to generate (locate) character candidates which are used as an input to [69]. In this way, we have the spatial information of the recognized characters. 3) Our previous work considers only bigrams to encode textual cues. In Chapter 4, various layers of textual cue encoding is performed (bi- and trigrams) and their effects are compared. 4) New analysis on the background removal and text saliency is provided on ICDAR13. 5) In depth analysis on textual and visual cues for fine-grained classification is provided. 6) Our previous work uses only BOW as visual features. Visual baseline is substantially improved using GoogLeNet features [160]. 7) Various fusion techniques are used to combine different modalities and their influences are discussed. 8) The proposed method is compared against state-of-the-art text detection methods [96, 128, 54] on text saliency, end-to-end character recognition and fine-grained classification. 9) The proposed method is applied on a new application ’Logo retrieval’.

4.2 Related Work

Text Detection. Text detection methods aim at automatically detecting and generating bounding boxes of words in natural scene images. Text detection methods can be categorized into two classes based on how they search character regions: a connected component [31, 128, 87, 191] and a sliding window approach [178, 179, 69]. Connected component approaches aim at segmenting characters using pixel similarities, e.g. contrast [82], stroke width [31] and intensity [126] whereas sliding window based approaches search the image over different scales and window sizes to locate character regions. For both methods, word candidates are detected by further verifying and combining the generated character candidates. To verify and combine character candidates, geometric, structural and appearance properties of text are derived from hand-crafted rules [31] or obtained by learning [178, 193, 135, 69, 189, 103].

All these methods extract geometric, structural and appearance features from candidate regions to verify if a region contains text or not. It is difficult to have one global parameter setting which would accommodate for all possible text variations in natural images [190]. Therefore, it is necessary to tune these parameters for every new alphabet, text style and size. In contrast, our approach focuses more on background connectivity rather than text regions. The proposed method does not extract text specific features. Therefore, it does not require any tuning for varying text size, style and orientation. Additionally, state-of-the-art methods combine characters into words by a learning or a rule based approach. However, the information loss at these steps are irreversible. In contrast, we use characters instead of words to represent textual information in the images. Recently, [68, 55] use similar ideas as in object proposals [167] but this time to generate a small set of word candidates. A reduced number of word candidates makes it possible to use more complex classifiers for word recognition. Such work can highly benefit from the proposed text detection method to reduce word box proposals even further.

Visual Saliency. The aim of visual saliency detection is to separate attention-driven regions and other regions (e.g. background) [138, 107]. In this way, the vast amount of incoming visual data (background) is eliminated. This helps to extract more reliable information because the background is eliminated. Therefore, it is widely used in image processing, for scene classification [155, 171], object recognition [82] and visual search [30].

Saliency for text detection has only recently received some attention [152, 20, 166, 159]. Text in natural scenes is typically designed to attract attention. In the experiment conducted by Judd et al. [76], it is shown that scene text fragments receive a high number of eye fixations (i.e. attention). Psychophysical experiments conducted by Wang and Pomplun [177] show that regardless of the text position, text features are more discriminative than generic saliency features. Recently, Jiang et al. [71] provide a large scale dataset for visual saliency. The authors measure saliency by following the mouse-tracking behavior of users. Interestingly, they observe that even though scene text is not explicitly defined as a category in MS COCO [97], scene text consistently attracts human attention. Other work by Shahab et al. [152] compares different saliency methods and concludes that scene text is the most salient. Recently, [96] proposes a text detection method which relies on text saliency. The method uses a Bayesian framework and integrates visual cues tailored for text detection to obtain text saliency. All this research shows that text in natural images is salient and therefore we rely on non-salient regions as our primary cue for background detection.

Existing methods for text detection using visual saliency [152] mainly focus on bottom-up information such as edges, corners, color distinctiveness and lines of symmetry. Top-down models are task dependent and use saliency and context to steer the search for objects in images. This process is inspired by human focus-of-attention mechanisms. For instance, while searching for text, humans will focus on road signs, commercials and billboards rather than other areas [76]. Torralba et al. [163] use context information to fixate locations of targeted objects (e.g. pedestrians). Our approach is inspired by Wei et al. [181]. The authors use boundary priors to steer the search for salient object detection. In our approach, background information is used for text detection.

Multimodal Fusion. The use of (textual) captions and visual information for video classification has extensively been studied. We refer to [104] for an overview. Moreover, textual in combination with visual cues have also been used for document image analysis [3, 145, 33]. Others [26, 144] combine visual and textual information to recognize logos and stamps in documents. However, the use of automatically extracted text information from natural images for scene classification has largely been ignored. Text is fused with visual cues for scene classification by Wang et al. [176]. The method uses Flickr images and their associated social tags.

Others [199], propose to combine visual features extracted from the surroundings of text regions with visual features from the full image. In contrast, we propose to use recognized text from images in combination with visual features.

Similar to our approach, [165] proposes to use scene text in combination with visual features to improve book spine recognition. However, our method focuses on combining textual and visual features for fine-grained classification and logo retrieval.

4.3 Background Removal

Text can appear on unknown background with unknown text size, style and orientation in natural scene images. It is difficult to have one global parameter setting which would accommodate for all these variations in text [190]. Therefore, we tackle the problem of detecting text from a different point of view. Rather than asking "what is the property of scene text", we ask the question "what is the property of scene background". Keeping in mind that scene text is highly contrasted with background, answering this question would also reveal the location of scene text. As a result, the proposed method would not require any tuning for varying text size, style and orientation. Moreover, eliminating background to infer text location has additional benefits, *(i)* the search space is reduced allowing the extraction of computationally intensive features for character recognition, *(ii)* background clutter is removed reducing false text detections/recognitions.

Background pixels are mostly homogeneous within themselves e.g. fences, boards, roads, buildings, windows etc.. This homogeneity allows defining connectivity between background pixels. Moreover, text is designed to attract attention [31, 128, 166, 177] and usually strongly contrasts with background. Hence, text boundaries usually correspond to strong intensity changes. Therefore, we propose to select initial background seeds from non-salient pixels and grow these seeds using connectivity of background. These seeds will grow until strong intensity changes are reached e.g., text/background transitions. Background seeds form connectivity between all pixels except those that belong to the text regions. An illustration of seed growing is shown in Fig. 4.3a. Blue dots represent initial background seeds whereas red lines represent the connectivity path formed by these initial seeds (blue dots).

To form connectivity between background pixels, we use conditional dilation (δ). Conditional dilation is a basic dilation which is conditioned by a mask image (i.e., the gray-level image I in our case). The conditioning is obtained by defining the output as the intersection of the dilation and I, formulated by:

$$\delta_I(J) = (J \oplus S) \wedge I \ , \tag{4.1}$$

where $J \oplus S$ stands for the dilation of J (the image consisting of only background seeds) and S (the structuring element 3-by-3 square), and $\wedge$ denotes the element-wise minimum.

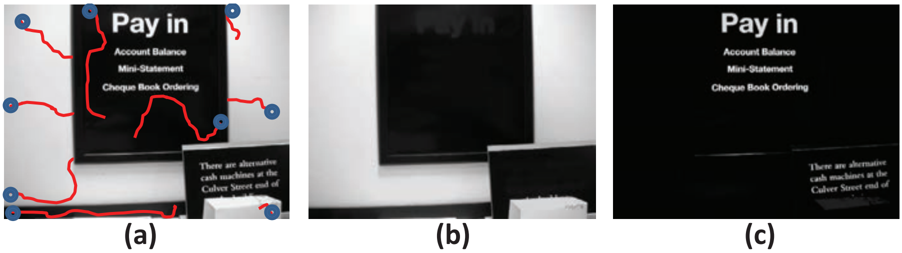


Figure 4.3: (a) Original image and selected initial background seeds (blue dots). The connectivity path formed by these initial seeds is represented by the red dots (b) recovered background and (c) background removed image regions [83].

To obtain a reconstructed background image (ρ) of image I, given the image consisting of the initial background seeds, J_0, eq. 4.1 is executed until stability is reached. That is, starting from the initial background seeds J_0 repeat $J_n = \delta_I(J_{n-1})$ until $J_n = J_{n-1}, (n = 1, 2, 3...)$ and obtain ρ by $\rho = J_n$.

The selection of background seeds J is essential for background detection process. The proposed method selects the seeds based on saliency and scene text priors as explained in the next section.

4.3.1 Background Seed Selection

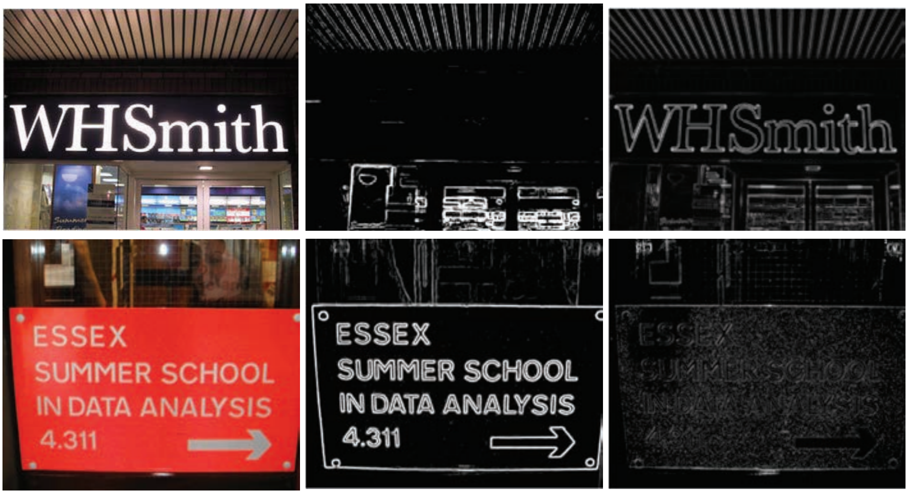
Color Saliency

In general, it can be assumed that color is homogeneous for many background regions such as roads, sky, buildings and so on. Moreover, color edges correlate with high contrasted text fragments. To exploit this, we propose to detect color edges by color boosting algorithm [169]. The method uses information theory to correlate gradient strength with information content.

To be precise, let $f_{o,x} = (O_{1x}, O_{2x}, O_{3x})^T$ be the spatial image derivatives in the x dimension where O_1, O_2, and O_3 stand for the opponent color channels. The information theory relates the information content of an event to its frequency or probability

$$I(f_{o,x}) = -\log(p(f_{o,x})) \;, \tag{4.2}$$

where I is the amount of information and $p(f_x)$ is the probability of the spatial x-derivative. According to information theory, rare events are more informative (i.e. higher information content). Consequently, [169] proposes to focus on rare color derivatives. A color saliency boosting function g is used to transform the vectors with equal information content to have equal influence on the saliency map. The distribution of image derivatives in opponent color space is characterized by a covariance matrix M. Eigenvector matrix U and an eigenvalue matrix V are obtained by the decomposition of matrix M. Then, the color saliency boosting function g is obtained by:



(a) (b) (c)

Figure 4.4: An example of saliency maps: (a) Original Images, (b) Color Boosting and (c) Curvature Shape Saliency. It is shown that colorful edge transitions are emphasized by color saliency while colorless edge transitions are emphasized by curvature saliency.

$$g(f_{o,x}) = V^{-1}U^{T}f_{o,x} \ . \tag{4.3}$$

Once g is determined, the color boosting saliency (S_c) is expressed by:

$$S_c = H(g(f_{o,x}), g(f_{o,y})) \ , \tag{4.4}$$

where H is the saliency function. Color boosting approach is used to enhance the saliency of colorful text/background transitions and to suppress background regions. In Fig. 4.4b, an example is shown of applying color boosting.

Curvature Saliency

Obviously, the color saliency measure is inappropriate for colorless edge transitions, see the top image in Fig. 4.4b. Consequently, in addition to color saliency, we aim for a shape-based saliency measure. To this end, we define curvature saliency (L) by:

$$L = \sqrt{f_{I,xx}^2 + f_{I,xy}^2 + f_{I,yy}^2} \ , \tag{4.5}$$

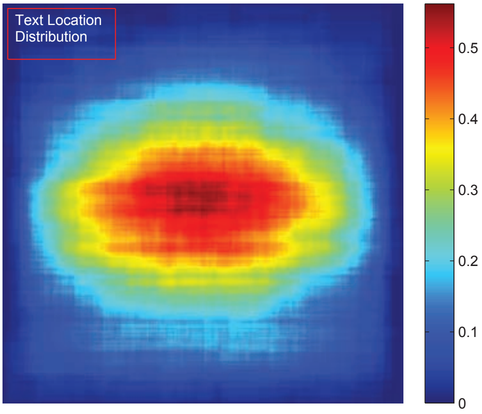


Figure 4.5: Location occurrence probability of text in the ICDAR03 training dataset. The probability distributions show that text rarely occurs at the image borders.

where $f_{I,xx}$ and $f_{I,yy}$ stand for the second-order derivatives of the intensity image $f_I(x, y)$ in the x and y dimensions, respectively. Due to contrast between text and its background, text regions result in high responses to curvature saliency even for colorless edge transitions.

Spatial Context

Spatial context is described by the likelihood of finding an object in a certain position [44]. It has been shown to be beneficial to distinguish objects in the scene [44]. The proposed method also uses spatial context. To this end, text location priors are used to obtain background location priors. The proposed method treats background and text pixels as figure/ground pixels. To this end, the location occurrence probability of text is computed for the ICDAR03 [105] image training set. The occurrence probability for a given location is computed by counting the frequency of text for that particular location for the full training set. The text-location occurrences shows that text regions are more in the center of the image, see Fig. 4.5. Text can also be placed off the center, however they rarely touch image borders. Image borders usually consist of background such as sky, road and grass [44].

The ICDAR03 dataset mostly consists of images where text is in focus. However, the text location prior also holds for other datasets. Recently, a text detection dataset which consists of 67K images is collected by [175]. The images are from complex everyday scenes. Unlike ICDAR datasets, the images were not collected with text in mind and thus they contain a broad variety of text instances [175]. The authors compared the text distributions of the new dataset and the existing ICDAR datasets. They show that the Coco-Text dataset has a more uniform text distribution.

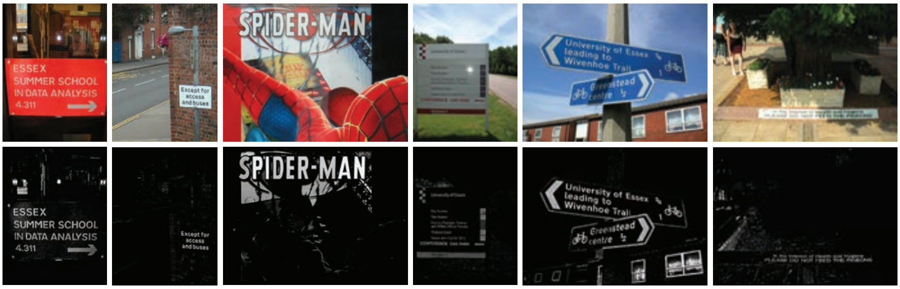


Figure 4.6: Original images and text saliency maps obtained. Most of the background regions are filtered out while text regions are preserved by the proposed method. The proposed method is robust against photometric changes e.g., shadow and highlights, text size, style and orientation.

However, image borders still rarely contain text. Therefore, our observation about image borders that they usually contain non-text pixels also holds for the Coco-Text dataset (uncontrolled text detection dataset). To this end, pixels at the image borders are used as initial background seeds. Salient regions which are connected to image borders in color and curvature saliency maps are suppressed using eq. 4.1. Pixels at image borders are used as initial background seeds, and the original color and curvature saliency maps are used as intensity images.

4.3.2 Background Detection and Text Saliency

The refined color and curvature saliency maps are normalized to a fixed range $[0,1]$ and linearly combined. Regions which do not have any response on this combined saliency map are considered as final background seeds. The background of the input image is constructed using these final background seeds and eq. 4.1. Text saliency map is obtained by subtracting the background from the input image. The proposed method outputs a text saliency map which provides information about how likely a region contains text (See Fig. 5.3 for an illustration). This saliency map is further processed to extract textual cues.

4.4 Textual Cue Encoding

Text saliency obtained in Section 4.3 is used to perform character recognition to extract textual cues. Two different approaches are used to perform character recognition. First, text saliency map is directly fed into leading commercial OCR engine (ABBYY). Second, text saliency map is further processed to obtain character candidates. Then, character candidates are fed into a state-of-the-art text recognition algorithm [69].

4.4.1 Character Recognizer - ABBYY

We first use ABBYY, leading commercial OCR engine, to perform character recognition on the text saliency. ABBYY receives an image as input and outputs recognized characters within that image. The gray level text saliency map is used as an input to ABBYY (no binarization is required). The recognized characters by ABBYY are directly used for textual cue encoding. The output of recognized characters are used to form bi- and trigrams without considering their spatial relations.

4.4.2 Character Recognizer - Flexible

Location information of the recognized characters cannot be obtained from ABBYY which restricts to spatially constrain textual cues. Moreover, remarkable performance improvements on character recognition algorithms allows to perform more reliable textual cue encoding. Therefore, in addition to ABBYY, we use state-of-the-art character recognition algorithm [69] to perform textual cue encoding.

Character Localization

The proposed method generates a gray level text saliency map. To extract textual cues, we need to know where the text is. However, the generated saliency map does not provide bounding boxes to explicitly locate character areas. Therefore, a character localization method is applied on text saliency map.

A segmentation (binarization) based algorithm is used to locate text regions. We use the binarization algorithm from our earlier work [42]. The method is efficient and proven to work well for natural scene images. The method uses statistics of image intensities. It models the distribution by a generalized extreme value theory to determine the threshold for binarization. We refer [42] for the details of the algorithm. Each connected component generated after binarization process is considered as a character candidate.

The proposed method is not restricted to a segmentation algorithm. A sliding-window approach can also be used to locate text with an exhaustive search in spatial and scale space. However, a sliding-window approach generates a large number of candidate regions which need to be processed by a recognizer. This increases the computational time. Therefore, a segmentation based approach is used to locate text.

Character recognition

The character candidates generated by the approach outlined in Section 4.4.2 are used as input of a character recognition algorithm [69]. The method uses a four layer convolutional neural network. The network takes as input a gray-scale image, resizes it to 24×24 pixels, and normalizes the image by subtracting the patch mean divided by the standard deviation (i.e. whitening). The output is a probability $p(c|x)$ for the Latin alphabet, digits and a non-text class resulting in a total of 37 classes. These character probabilities are obtained by feeding last channels of the network into a soft-max manner.

Spatial Constraints

Recognized characters in Section 4.4.2 are used to encode textual cues. However, the rich semantics of text cannot be fully conveyed by using only single characters. Character, 'P', 'E', 'A', 'T', can be combined into 'TAPE', 'PATE', or 'PEAT'. Therefore, rather than single characters, we propose to encode textual cues by forming bi- and trigrams. However, it is still not possible to distinguish between 'TAPE', 'PATE' and 'PEAT' by considering all possible bi- and trigram combinations of characters forming these words. Accordingly, we propose to encode textual cues with spatial constraints. For instance, recognized characters of the word 'TAPE' will form bigrams 'TA', 'AP', 'PE', 'TP' and 'AE' whereas the word 'PATE' will form 'PA', 'AT', 'TE', 'PT' and 'AE'.

To spatially constrain bi- and trigrams, we propose an approach inspired by [92]. However, we do not define exact image boundaries to localize information. We allow more general feature grouping by considering the following pairwise relations: *(i)* The ratio of the heights of two characters *(ii)* The angle between two character centers *(iii)* Euclidean distance between two consecutive character centers *(iv)* Shared area between two characters. Thresholds are set according to state-of-the-art text detection algorithms [31, 127, 37, 96] which form textlines between characters using the above constraints. The proposed method establishes connections between character candidates which satisfy the spatial constraints. These connections help to form bi- and trigrams. Then, textual representations are obtained by forming histograms of character combination occurrences. Each representation is independently normalized (L_1) and concatenated to obtain the final textual cue.

The proposed textual cue representation has certain benefits. First, it is not possible to distinguish anagrams (e.g. TAPE and PEAT) by considering all possible combinations of the characters forming the words. Therefore, the spatial constraints help to preserve the ordering of the characters. Second, the proposed method explicitly avoids word detection to extract textual cues. The word formation step causes information loss due to introduced ad-hoc rules. If the connection between characters of the word is broken due to ad-hoc rules or missed characters, the proposed method can still extract textual cues. For instance, if the character "A" of "TAPE" is missed, word formation step would most likely not detect the word "TAPE". However, the proposed textual cue encoding would still encode "TP" and "TPE" (See Fig. 4.7 for an illustration).

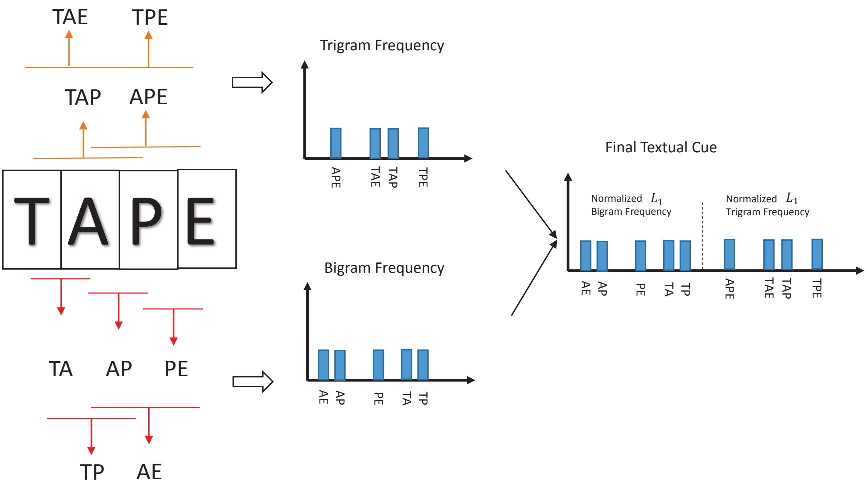


Figure 4.7: An illustration on how the spatial constraints are used to encode textual cues. Each box represents a character candidate. These character candidates are used to form bi- and trigram representations. Bi- and trigrams are formed only when the spatial constraints are satisfied.

4.5 Fusing Textual and Visual Information

Extracted textual and visual features are fused using three techniques, namely, early, late and kernel fusion. In this way, the relations between different modalities (i.e. text and visual features) are exploited at various levels of abstraction.

Early Fusion Early fusion is performed at the feature level. After textual and visual features are extracted, the two modalities are concatenated in a single feature vector. Then, the Support Vector Machine (SVM) is used to classify this combined feature vector. Early fusion benefits from learning the regularities formed by the components independently based on different modalities. However, it restricts the choice of classifier and/or kernel to be the same for different modalities.

Late Fusion Late fusion is performed at the decision level. After textual and visual features are extracted, SVM classifiers are trained on the unimodal features independently. A prediction is obtained for each modality. Then, these predictions are combined by averaging. In contrast to early fusion, various classifiers can be trained according to the modalities. On the other hand, late fusion does not exploit the feature level correlation among modalities. Moreover, the learning process for late fusion is more time consuming than early fusion since each modality

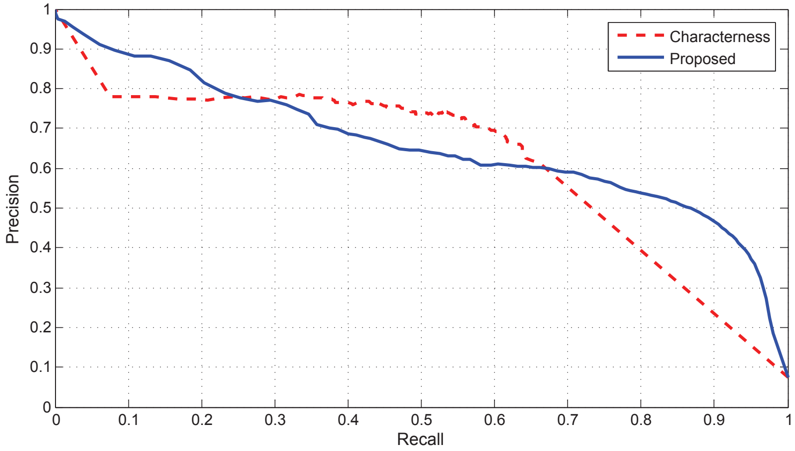


Figure 4.8: Average precision-recall curves on ICDAR13 dataset. The proposed method (blue line) is fully unsupervised and achieves higher recall (after 0.7) with higher precision than Characterness [96] (red dashed line) which is a supervised text detection method.

requires training a different classifier.

Kernel Fusion Kernel fusion is performed at the kernel level. After the textual and visual features are extracted, kernels are constructed on the unimodal features. Then, these kernels are combined by using the sum operation [4].

In this chapter, we explore the above fusion techniques to exploit the relations between modality components for fine-grained image classification.

4.6 Validation of Text Saliency

The proposed method provides a text saliency map. [96] is similar to our approach because it generates a text saliency map too. Moreover, [96] is a supervised approach which outperforms other object saliency methods on text detection on ICDAR13 [84]. To demonstrate the effectiveness of the proposed method, we compare our results with [96]. The evaluation metric is used as in [96]. The pixel level annotations of the ICDAR13 are used as ground truth. The dataset consists of 229 images. For comparison reasons, the same set of randomly selected 100 images in [96] is used for testing.

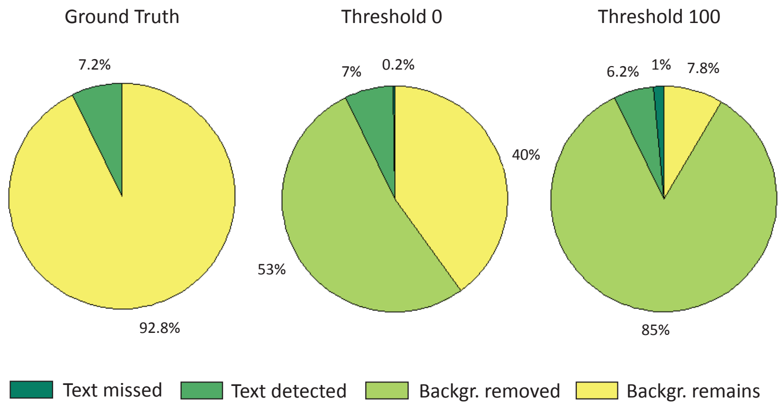


Figure 4.9: The number of background and text pixels for the ICDAR13 dataset. The majority of the scenes consists of background pixels (93%). The text saliency map is normalized to the range of [0,255]. Different thresholds indicate the number of text/background after the proposed method is applied. The results at threshold 0 shows that proposed method suppresses a large amount of background (53%) while preserving most of the text regions. The amount of suppressed background reaches (85%) whereas the amount of missed text detections is still reasonable at threshold 100.

4.6.1 Experiments and Results

We evaluate the effectiveness of our text saliency method based on the precision-recall (PR) curve as proposed by [96]. The generated text saliency map is normalized to the range of [0,255]. Then, the PR curve for an image is computed by binarizing the saliency map using thresholds varying from 0 to 255. A full PR curve is generated by averaging PR curves over all test images.

The results are shown in Fig. 4.8. The proposed method (blue line) is fully unsupervised and achieves higher precision than [96] (red dashed line) at higher recall regions.

To provide insights in the background removal process, we measure the total amount of background/text and the amount that remains after the proposed method has been applied. It is shown in Fig. 4.9 that 93% of the pixels of the overall images consists of background. We represented that the proposed method reaches higher precision values at higher recall regions compared to [96]. The reason is that the proposed method focuses on removing connected background pixels rather than the detection of text regions. The algorithm benefits from the background connectivity process to remove larger number of pixels. At the same time, the proposed method avoids to reach the text regions (due to contrast). Hence, the proposed method eliminates more background and misses a limited number of text pixels. Fig. 4.9 illustrates that a large number of background pixels are suppressed (53%) by the proposed method (at Threshold 0) while only a limited number of text pixels are missed (0.2%). The proposed method is suited as pre-

Method$\downarrow$	Cl. Rate (%)
Orig. Image + ABBYY	37
Backg. Removed (Proposed) + ABBYY	**62**

Table 4.1: The impact of background removal on end-to-end character recognition on ICDAR03. ABBYY is a commercial OCR engine. *Orig.Image + ABBYY* uses original image whereas *Proposed+ABBYY* uses text saliency map (background removed input images) as input for ABBYY character recognition. These approaches differ from each other due to background removal. This shows that eliminating the background from the images increases the OCR accuracy.

processing step for text detection algorithms to reduce the search space (e.g. [68, 55]). The rest of Chapter 4 uses [42] to binarize the text saliency map. Systematically changing the threshold values for binarization is only used to obtain PR curve as in [96].

4.7 Character Recognition Evaluation

4.7.1 Dataset

End-to-end character recognition performance of our proposed method is evaluated on the publicly available ICDAR03 test dataset. The dataset consists of 5370 letters in 249 images. The dictionaries supplied by the dataset are not used to refine the recognition results.

4.7.2 Experiments and Results

We conduct two experiments. First, we evaluate the effect of background removal on the character recognition accuracy. Second, we compare the results for end-to-end character recognition performance of our method against the state-of-the-art text detection methods [96, 54, 128].

Experiment I. We compare *Proposed+ABBYY* against *Orig.Image+ABBYY*. For *Orig.Image+ ABBYY* character recognition, the input image is directly fed into ABBYY without any processing. *Proposed+ABBYY* also uses ABBYY for character recognition. However, text saliency map is used as input. The results are shown in Table 4.1. Although removing the background from the images may decrease the character detection recall in some cases, the experimental results show that background removal increases the overall character recognition recall of ABBYY by 25%. This is because traditional OCR systems are designed to work on documents with homogeneous backgrounds. Their performance deteriorates for natural scene images with cluttered and inhomogeneous background [116].

Experiment II. End-to-end character recognition results of our proposed methods are compared against state-of-the-art text detection methods [54, 96, 128]. Table 4.2 summarizes the results.

Method↓	Cl. Rate (%)
Text Detection [54] + Text Recog. [69]	38
Text Detection [96] + Text Recog. [69]	53
Text Detection [128] + Text Recog. [69]	64
Text Detection (Proposed) + Text Recog. (ABBYY)	62
Text Detection (Proposed) + Text Recog. [69]	**79**

Table 4.2: End-to-end character recognition performance of our proposed methods against text detection methods [54, 96, 128] on ICDAR03. [69] is used for character recognition for all the methods. The methods only differ in character detection steps. The results show that proposed method significantly outperforms [54, 96, 128] on end-to-end character recognition on ICDAR03. *Proposed+ABBYY* and *Proposed+[69]* differ in character localization and recognition steps. The results show that *Proposed+[69]* outperforms *Proposed+ABBYY*.

We detect character candidates using state-of-the-art text detection algorithms [54, 96, 128]. [69] is used to perform character recognition. End-to-end character recognition performance obtained using character candidates of the proposed method significantly outperforms (15%) other methods. Additionally, *Proposed+ABBYY* and *Proposed+[69]* differ in character candidate generation and character recognition steps. *Proposed+[69]* (described in Section 4.4.2) uses the state-of-the-art character recognition method [69] rather than ABBYY as in *Proposed+ABBYY* (described in Section 4.4.1). A significant improvement (17%) over *Proposed+ABBYY* is obtained. This indicates that *Proposed+ABBYY* is restricted by the accuracy of ABBYY. *Proposed+[69]* reaches state-of-the-art end-to-end character recognition accuracy up to 79% on this dataset.

Efficiency. The experiments are conducted on a laptop (Intel(R) Core(TM) i7-4810MQ Processor (2.80 GHz)) using Matlab. To run the method on a 480×640 resolution image takes 0.1s for each color and curvature saliency map extractions, 0.12s for recovering the background, 0.1s for binarization and 0.001s for each character candidate recognition.

4.8 Fine-grained Classification

4.8.1 Dataset and Implementation Details

Dataset. We use sub-classes of the ImageNet[‡] *building* and *place of business* sets to evaluate our fine-grained classification based on textual and visual information. The dataset consists of 28 categories with 24,255 images, see Fig. 4.10 for the list of categories. In the experiments, we use all images from these categories. Note that many images may not necessarily contain scene text fragments. The number of images that contain text varies e.g. *Bistro* (21%) and *Dry Cleaner* (89%).

[‡] http://image-net.org/

Textual Cue Encoding↓	Performance (mAP%)
unconstrained bigrams	13.1
unconstrained trigrams	12.6
Proposed spatially constrained bigrams	24.8
Proposed spatially constrained trigrams	24.0
Proposed spatially constrained [bi+tri]grams	**28.4**

Table 4.3: The influence of different textual cue encodings on fine-grained classification performance (mAP). Encoding textual cues as bi- or trigrams produce similar results. Adding proposed spatial constrains on textual cue encodings significantly outperforms the version without spatial constrains. Combining bi- and trigram representations outperforms each individual representation.

Implementation notes We use average precision as the performance metric. We repeat all experiments three times to obtain standard deviation scores to validate the significance of the results. We use three types of features as visual baseline *1.* a standard bag of visual words (BOW) approach with SIFT using 4000 words with 1×3 and 2×2 spatial pyramid, *2.* we use a pre-trained model of GoogLeNet [160]. The GoogLeNet is trained on ILSVRC 2012, provided by the Caffe library [§]. We use the last average pooling layer as features (1024-dim), which are further normalized to unit lengths (DEEP), *3.* GoogLeNet is fine-tuned with a 28-way softmax classifier. The learning rate is 0.001, decreased by a factor of 10 every 5 epochs. The weight decay is set to 0.0005 and the momentum is 0.9. The network is fine-tuned for 20 epochs (DEEP-FT). We use the histogram intersection (for BOW) and linear (for DEEP and DEEP-FT) kernels in Libsvm and use its default value for the C parameter (=1) without any tuning. For text classification, we use the proposed method based on unconstrained and spatially constrained bi- and trigrams with the histogram intersection kernel with the same settings. The visual and textual modalities are fused using techniques described in section 4.5.

4.8.2 Experiments and Results

We perform six experiments. First, we evaluate the influence of different textual cue encodings on the final classification performance. Second, we quantify the impact of textual cues for fine-grained classification. Third, we compare our results against state-of-the-art text detection methods for textual cue extraction. Fourth, we assess the influence of three different fusion strategies. Fifth, we quantify the complementarity of the extracted features. Sixth, we discuss the influence of the performance change with respect to the amount of text in images.

Experiment I. We evaluate the following textual cue encodings: (1) unconstrained bigrams, (2) unconstrained trigrams (3) spatially constrained bigrams, (4) spatially constrained trigrams, and (5) spatially constrained bi- + trigrams. The results are summarized in Table 4.3. Represent-

[§] https://github.com/BVLC/caffe/tree/master/models/

Source of Info. ↓	Performance (mAP%)
Textual-only	28.4
Visual-only (BOW)	34.9
Visual-only (DEEP)	53.3
Visual-only (DEEP-FT)	60.3
Textual + Visual (BOW)	**47.9**
Textual + Visual (DEEP)	**66.2**
Textual + Visual (DEEP-FT)	**70.7**

Table 4.4: The impact of textual and visual cues for fine-grained classification results (mAP). The results show that textual cues extracted by the proposed method achieve limited accuracy, 28.4%. The low performance of visual-only shows that visual information is not sufficient. Combining textual and visual cues significantly outperforms visual-only results with 13%, 12% and 10.4% for BOW, DEEP and DEEP-FT respectively. This shows that textual information is beneficial for fine-grained classification and is complementary to the visual cues. The gain in the performance is almost the same for three different visual-only baselines, even though their performance are at different ranges. This is due to the fact that textual and visual cues are from completely different sources.

ing textual cues in terms of bi- or trigrams produce similar results. However, the combination outperforms each individual representation. Finally, the significant performance improvement between spatially constrained and unconstrained textual representations indicates that spatially constraining textual representations increases discriminative power of textual cues.

Experiment II. To quantify the influence of visual and textual cues, we compare the results of visual-only (BOW), visual-only (DEEP), visual-only (DEEP-FT), textual-only, textual+BOW, textual+DEEP and textual+DEEP-FT. The classification scores per category are shown in Fig. 4.10 and mAP is given in Table 4.4. The approach using textual cues, extracted by the proposed method, achieves an accuracy of 28.4%. The baselines using visual-only features achieve 34.9%, 54.5% and 60.3% for BOW, DEEP and DEEP-FT respectively. Using more discriminative visual features (DEEP-FT) significantly improves the visual baseline of BOW (26%). Using textual cues in combination with visual cues increases the mean average precision up to 47.9%, 66.2% and 70.7% for BOW, DEEP and DEEP-FT respectively. The performance gain due to combining visual and textual cues is preserved, 13%, 12% and 10.4% for BOW, DEEP and DEEP-FT respectively (even though their individual performance are substantially different). Hence, textual information is beneficial for fine-grained classification and is complementary to visual cues.

Adding textual information to visual cues improves the accuracy of 26 out of the 28 classes. The low accuracy for textual cues compared to visual cues can be explained by the lack of scene text in many images, as is the case for the *Bistro* class. Nevertheless, text cues outperform visual cues for *Discount House*, *Steak House*, *PawnShop*, *Cafe* and *Dry Cleaner*. Intra-class variation for *Discount House* and *Steak House* are high. Therefore, the images within these classes are

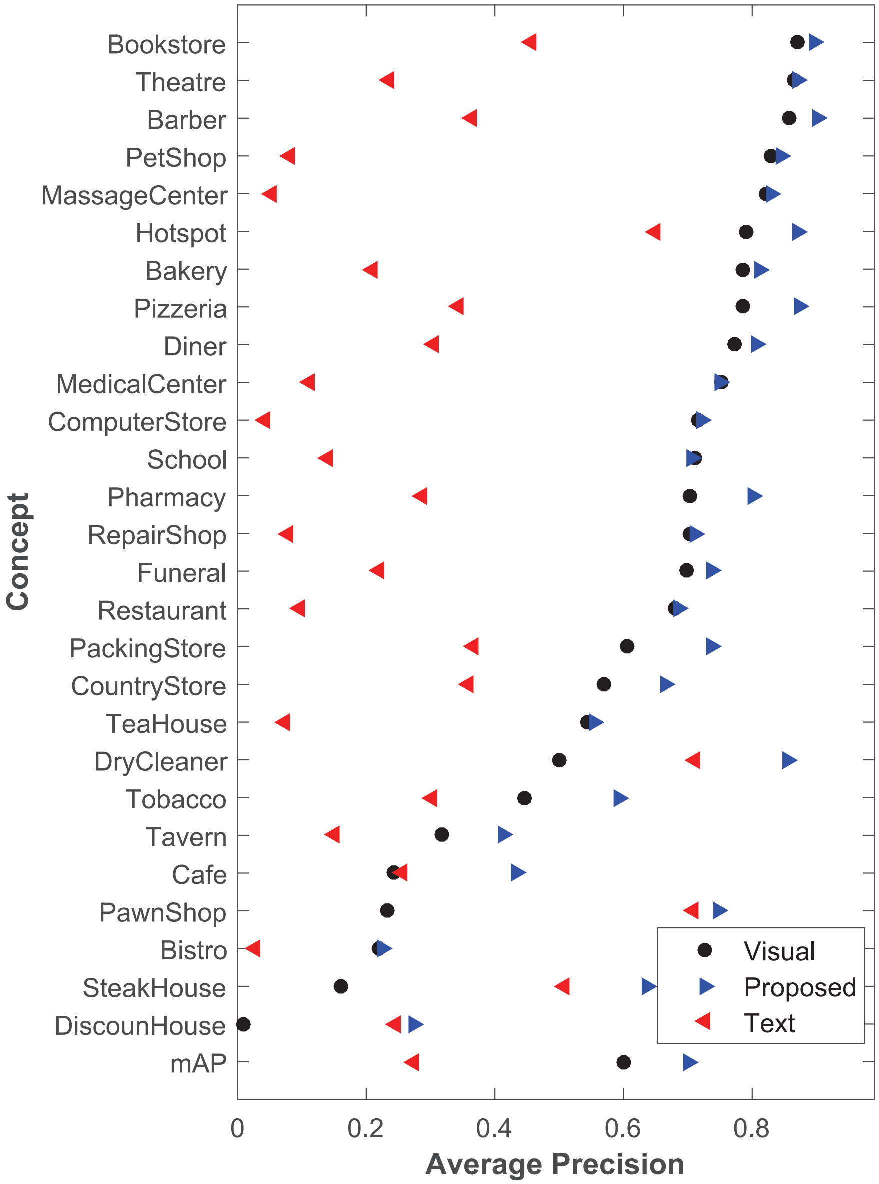


Figure 4.10: Fine-grained classification results on each class for visual-only (DEEP-FT), textual-only and proposed method. The visual mAP is 60.3 ± 0.2, text is 28.4 ± 1.7 and proposed is 70.7 ± 0.6. Adding textual cues significantly outperforms (10.4%) visual-only results.

Method ↓	Performance (mAP%)
Textual-only (Text Detection [54])	10.9
Textual-only (Text Detection [128])	17.8
Textual-only (Text Detection [96])	19.9
Textual-only (Text Detection Proposed)	**28.4**
Textual (Text Detection [54]) + Visual (DEEP-FT)	63.0
Textual (Text Detection [128]) + Visual (DEEP-FT)	66.4
Textual (Text Detection [96]) + Visual (DEEP-FT)	67.6
Textual (Text Detection Proposed) + Visual (BOW)	47.9
Textual (Text Detection Proposed) + Visual (DEEP)	66.2
Textual (Text Detection Proposed) + Visual (DEEP-FT)	**70.7**

Table 4.5: Comparison of fine-grained classification results of textual cues using the proposed text detection method and state-of-the-art text detection methods [96, 54, 128]. *Textual* methods differ in text detection. Textual cues are extracted for all the methods as proposed in this chapter. [69] is used to perform character recognition. Bi- and trigrams are formed between the recognized characters by considering the proposed spatial pairwise constraints. The proposed method significantly outperforms [96, 54, 128] for textual-only cues and also visual+textual cues.

visually dissimilar and are difficult to group them together without text (even for humans).

Experiment III. The proposed method is compared against the state-of-the-art text detection methods [96, 54, 128] for textual cue extraction. The methods only differ in character detection. The same steps are followed for all the methods to extract textual cues as proposed in this chapter. Table 4.5 illustrates that textual cues extracted by the proposed method outperforms [96, 54, 128]. Moreover, we compare our previous result in [83] for which we encode the textual cues using unconstrained bigrams, ABBYY for character recognition and BOW for visual features. Our previous result is 39%. *Textual(Proposed)+BOW* outperforms it significantly (9%). This shows the importance of *1.* spatially constraining bigrams, *2.* adding another layer of encoding (trigrams), *3.* high character recognition performance on fine-grained classification. Additionally, using more discriminative visual features [160] significantly improves the visual baseline. Consequently, the proposed method outperforms [83] by a large margin (32%).

Experiment IV. We assess the influence of the three different fusion strategies. These strategies are outlined in Section 4.5. Late fusion is performed in three different ways (*i*) predictions from each modality are combined equally using a sum operation (*sum*), (*ii*) another classifier is trained on top of predictions of the modalities to obtain weights to sum predictions from different modalities (*weighted sum*) and (*iii*) predictions from different modalities are added using a product operation (*product*). Table 4.6 shows the results. *Early* performs slightly better than the rest. Early fusion benefits from learning the correlation between features of textual and visual modalities. *sum* and *weighted sum* performs in a similar way and better than *product*. Kernel fusion performs slightly lower than early fusion. However, *Kernel* is still beneficial since it does not restrict the choice of kernel to be the same for different modalities. Therefore, *Kernel*

Fusion Strategy ↓	Performance (mAP%)
Early	**71.0**
Late - sum	70.0
Late - weighted sum	69.8
Late - product	65.0
Kernel	70.7

Table 4.6: Comparison of different strategies for multimodal information fusion.

is used for comparisons.

Experiment V. In this experiment, we explore the complementarity of the extracted features. All combinations are performed using kernel fusion strategy. The results are summarized in Table 4.7. Textual features extracted by different methods only differ in detected text regions. Beside that, textual cues are obtained in the same way. Therefore, combining textual cues together does not influence the overall accuracy. This also indicates that the proposed method covers all the information that [96, 54, 128] carries and more. BOW and GoogLeNet visual baselines are extracted using different features. Therefore, combining BOW and DEEP/DEEP-FT still improves DEEP/DEEP-FT only. This indicates that BOW and DEEP/DEEP-FT visual features are useful for fine-grained classification on this dataset. Combining all the extracted features still improves with 1% over the best result obtained in previous experiment (70.7%). Moreover, combining textual cues still significantly improves the performance even after combining all visual features. This indicates complementarity of visual and textual modalities.

Experiment VI. In this experiment, we discuss the performance change with respect to the amount of text in images. To this end, we have annotated text regions (as word bounding boxes) for the first 10 classes of the Con-Text dataset (in alphabetical order). All the text (Latin alphabet) visible and recognizable has been annotated. The annotated dataset consists of 9131 images. 5219 of these images contain at least one word box. In total there are 27601 word boxes annotated. We report per-class, *1.* the total number of images, *2.* the number of images with text, *3.* the percentage of images with text with respect to the total number of images, *4.* fine-grained classification rates (mAP) using only textual information (see Table 4.8).

"Dry Cleaner" obtains the maximum classification rate (71%). Hence, there is a strong correlation with the amount of text and the fine-grained classification rate since "Dry Cleaner" has the maximum number of images with text and also a high text percentage (89%). This holds also for the other classes which contain a high number of images with text, and a high text percentage (e.g. BookStore, Country Store). The classification rate is low when the number of images with text and text percentage is limited (e.g. Bistro, ComputerStore). The text percentage alone is not a conclusive indicator for high classification rate (e.g. DiscountHouse). It is necessary that there is enough data (number of images with text) to learn from. Moreover, overlapping text is also important. For instance, "Country Store" has a high number of images and text percentage. However, the classification rate is relatively small compared to "Dry Cleaner". This is due to the

Source of Feat. ↓	Performance (mAP%)
Textual-only (Text Detection [54])	10.9
Textual-only (Text Detection [128])	17.8
Textual-only (Text Detection [96])	19.9
Textual-only (Text Detection Proposed)	28.4
Textual-only (Proposed+[54]+[128]+[96])	28.4
Visual-only (BOW)	34.9
Visual-only (DEEP)	53.3
Visual-only (DEEP-FT)	60.3
Visual-only (BOW + DEEP)	55.8
Visual-only (BOW + DEEP-FT)	62.9
Visual-only (DEEP + DEEP-FT)	62.1
Visual-only (BOW + DEEP + DEEP-FT)	63.5
Visual (BOW + DEEP + DEEP-FT) + Textual (Proposed)	**71.6**

Table 4.7: The impact of combining the same and different modalities on fine-grained classification performance. Textual cues are extracted in the same manner for *Textual(Proposed)* and *Textual([96])*. They differ only in detecting text regions. Combining them does not influence the overall accuracy. Visual cues are extracted using different features. Therefore, combining BOW and DEEP/DEEP-FT still improves DEEP/DEEP-FT only. The largest gains are obtained when different modalities are combined.

diversity of text descriptions.

4.9 Logo Retrieval

In logo retrieval, the aim is to find all images of a query logo in an image collection, e.g., *Starbucks*. Logos may consist of text alone or text is an important part of the logo itself such as *Starbucks*, *Ford*, *FedEX* and *Google*. However, in logo retrieval, recognized text in natural scene images has never been exploited before [141, 142, 161].

4.9.1 Dataset and Implementation Details

Dataset. Our approach is validated on *FlickrLogos-32* [143]. The dataset consists of 32 brand logos, e.g., *Texaco*, *Pepsi* and *Google* and 30 queries per logo resulting in a total of 960 queries. The search set contains 40 images per logo and 3000 non-logo images.

Implementation notes. Again, we use average precision as the performance measure. To represent the visual cues, a standard bag of visual words (BOW) approach is used with a visual vocabulary of size 1 million. This visual representation reaches a retrieval accuracy of 54.8% mAP. This accuracy corresponds with what has been reported in [142]. To represent textual cues, we use spatially constrained bi- and trigrams and their combinations. The character candidates

Class↓	Total Num. Images	Num. Images with Text	Text Percentage (%)	Performance (mAP%)
Bakery	1214	467	38	21.19
Barber	1573	635	40	35.81
Bistro	287	59	21	3.12
BookStore	1333	724	54	44.46
Cafe	839	438	52	25.80
DryCleaner	1195	1065	89	71.46
ComputerStore	287	142	49	5.38
CountryStore	1224	866	71	35.01
Diner	1136	781	69	30.95
DiscountHouse	43	42	98	25.63

Table 4.8: The distribution of images with text and without text for the first 10 classes of the fine-grained dataset.

Textual Cue Encoding↓	Performance (mAP%)
bigrams	19.0
trigrams	18.9
[bi+tri]grams	**23.9**

Table 4.9: The influence of different textual encodings e.g., bi- and trigram and bi- +trigram in logo retrieval. Bi- and trigrams produce similar results. Combining bi- and trigram outperforms each individual representation.

within the query bounding boxes are kept for the query images to represent textual cues. The images are ranked by the cosine similarity between the normalized textual representations. To combine the visual and textual cues, we use late fusion.

4.9.2 Experiments and Results

We conduct two experiments. First, we quantify the influence of textual cues and the effect of different levels of textual cue encodings for logo retrieval. In the second experiment, we evaluate the complementarity of textual and visual cues for logo retrieval.

Experiment I. We evaluate different textual cue encodings: (1) bigrams, (2) trigrams and (3) bi- + trigrams. The results are summarized in Table 4.9. Representing textual cues as bi- or trigrams produces similar results. However, combining both representations outperforms the individual encodings. This implies that bi- and trigrams have similar discriminative power, yet they capture information at different levels. Therefore, they are complementary to each other.

Experiment II. To assess the influence of textual cues, we compare the results of textual-only, visual-only and textual+visual. The results are summarized in Table 4.10. The results show that the method based on textual cues extracted by the proposed method achieves an accuracy of 23.9%. Using textual and visual cues increases mAP up to 57.4%. This implies that textual

Source of Info. ↓	Performance (mAP%)
textual-only	23.9
visual-only	54.8
textual+visual	**57.4**

Table 4.10: Logo retrieval results (mAP) for textual-only, visual-only and textual+visual. The results show that the method based on textual cues extracted by the proposed method achieves an accuracy of 23.9%. Combining textual and visual cues increases the accuracy. This shows that textual information is beneficial for logo retrieval and is complementary to visual cues.

information is beneficial for logo retrieval. And that textual information is complementary to visual cues. Textual cues improve the retrieval accuracy of the logos which are formed by only text e.g., *esso*, *aldi* and *stella artois*. However, textual cues are not effective when the characters of the logos are all connected e.g., *fedex* and *ford*, when logos do not contain any text e.g., *apple* and *ferrari*, or when the characters are not in a common text font.

4.10 Conclusion

A method has been introduced to combine textual with visual cues for fine-grained classification and logo retrieval. While the state-of-the-art relies on visual cues only, in Chapter 4 we propose to combine recognized scene text and visual cues for fine-grained classification and logo retrieval. To extract text cues, we have proposed a generic, efficient and fully unsupervised algorithm for text detection. The proposed text detection method does not directly detect text regions but instead aims to detect background to infer text location. Remaining regions after eliminating background are considered as text regions. Then, text candidates have been processed by two methods to perform text recognition i.e. ABBYY commercial OCR machine and state-of-the-art character recognition algorithm [69]. Bi- and trigrams have been formed between the recognized characters by using proposed spatial encoding.

The proposed algorithm achieves state-of-the-art (end-to-end) character recognition accuracy on the ICDAR03. It is shown that bimodal information fusion of visual and textual cues increased the fine-grained classification accuracy by 10.4%. The proposed method outperforms state-of-the-art text detection methods [96] on text saliency and [54, 96, 128] on (end-to-end) character recognition and fine-grained classification. We improve earlier work [83] for fine-grained classification from 39.0% to 70.7% in *mAP*. Moreover, we applied our work also for logo retrieval. Textual cues proven to be complementary to visual cues for logo retrieval too. Combining textual and visual cues improves the logo retrieval performance over visual-only from 54.8% to 57.4%.

5

Scene Text for Image Classification and Retrieval[*]

5.1 Introduction

Fine-grained classification is the problem of assigning images to classes where instances from different classes differ slightly in the appearances e.g., flower types [133], bird [182] and dog species [100], and models of a product [108]. In contrast to coarse object category recognition e.g., cars, cats and airplanes, low-level visual cues are often not sufficient to make distinction between fine-grained classes. Even for human observers, fine-grained classification tasks usually require expert and domain specific knowledge. Accordingly, most recent works also integrated such domain specific knowledge into their solutions. For instance, dogs have ears, nose, body, legs etc., and the differentiation of dog species relies on the subtle differences in these parts. Different bird species have different wing and beak appearances, and such differences in local parts provide the critical information to categorize different bird types. [195, 184, 100] exploit the part information and extract features from particular parts for better birds and dogs recognition. In this chapter, we make use of the domain specific knowledge of *buildings*. We exploit the recognized text in images for fine-grained classification of building types. The building types studied in this work are places-of-businesses (e.g., bakery, cafe, bookstore etc.). Automatic recognition and indexing of business places will be useful in many practical scenarios. For instance, it can be used to extract information from Google street view images and Google Map can use the information to provide recommendations of bakeries, restaurants close to the location of the user.

Most of the time, the stores use text to indicate what type of food (pizzeria, diner), drink (tea, coffee) and service (drycleaning, repair) that they provide. This text information is helpful even

[*] S. Karaoglu, R. Tao, T. Gevers and A. W. M. Smeulders, "Words Matter: Scene Text for Image Classification and Retrieval," Pending revision in IEEE Transactions on Multimedia, 2015.

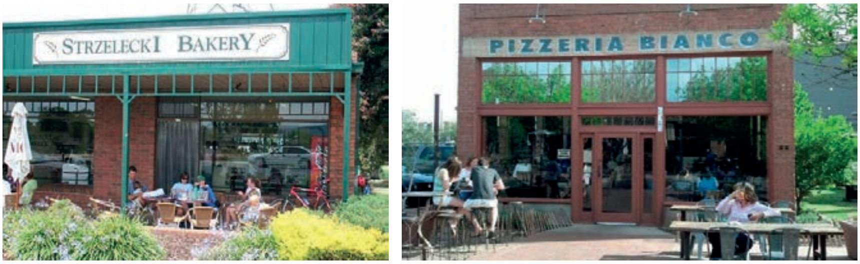


Figure 5.1: *bakery* and *pizzeria* example images. The two buildings are visually similar. Text can be used to differentiate the two shops.

for human observers to understand the content of the store. For instance, in Fig. 5.1, the images of two different buildings (*pizzeria* and *bakery*) have a very similar appearance. However, they are different types of business places. It is only possible with text information to identify what type of business places these are. Moreover, text is also useful to identify similar products (logo) such as *Heineken*, *Foster* and *Carlsberg*. Therefore, we propose a multimodal approach which uses recognized text and visual cues to do better fine-grained classification and logo retrieval.

The common approach to text recognition in images is to detect text first before they can be recognized [179, 69]. The state-of-the-art word detection methods [128, 178, 96, 183, 103] focus on obtaining a high f-score by balancing precision and recall. However, instead of using the f-score, our aim is obtain a high recall. A high recall is required because textual cues that are not detected will not be considered in the next (recognition) phase of the framework. Unfortunately, there exists no single best method for detecting words with high recall due to large variations in text style, size and orientation. Therefore, we propose to combine character candidates generated by different state-of-the-art detection methods. To obtain robustness against varying imaging conditions, we use color spaces containing photometric invariant properties such as robustness against shadows, highlights and specular reflections.

The proposed method computes text lines and generates word box proposals based on the character candidates. Then, word box proposals are used as input of a state-of-the-art word recognition method [67] to yield textual cues. Finally, textual cues are combined with visual cues for fine-grained classification and logo retrieval. The proposed framework is given in Fig. 5.2.

Chapter 5 has the following contributions. First, we combine word-level textual cues and visual cues for fine-grained classification and logo retrieval. In Chapter 4, we extract the textual cues at character-level. However, in this chapter, we propose to extract textual cues at word-level. The proposed method reaches state-of-the-art results on both tasks. Second, to extract the textual cues, a generic and computationally efficient word proposal algorithm which aims at high recall is proposed without any training involved. The proposed algorithm obtains state-of-the-art recall for word detection for a limited number of word box candidates. Third, contrary to what is widely acknowledged in text detection literature, we experimentally show high recall word detection is more important than high f-score for both applications considered in this work.

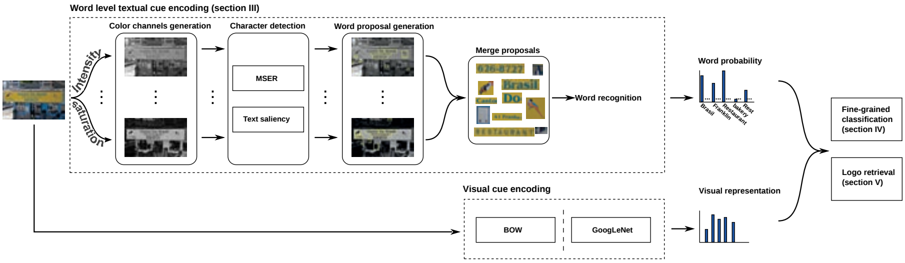


Figure 5.2: Pipeline of our multimodal approach. Text is encoded at a word level and utilized for fine-grained classification and logo retrieval. A generic and fully unsupervised word box proposal method is proposed to detect words in images. The method uses different color spaces and character detection algorithms (MSER [112] and text saliency [82]). The word box candidates are used as input for a state-of-the-art word recognition method [67] to perform word-level encoding. An English vocabulary consisting of around $90k$ words is considered [67]. For the visual cues, bag-of-words (BOW) and GoogLeNet features [160] are used. The multimodal approach combines the visual and textual cues.

5.2 Related Work

Word Detection. Word detection consists of computing bounding boxes of words in images. Existing word detection methods usually follow a bottom-up approach. Character candidates are computed by a connected component [31, 128] or a sliding window approach [178, 69, 179]. Candidate character regions are further verified and combined to form word candidates. This is done by using geometric, structural and appearance properties of text and is based on hand-crafted rules [31] or learning schemes [178, 69]. State-of-the-art word detection methods [128, 178, 96] focus on high f-score by the trade-off between recall and precision. Strict rules are used in character detection and word formation to keep only boxes that most likely contain words. As a consequence, methods aiming for high f-score may miss a number of correct word boxes. In contrast, we propose to generate word boxes with the goal to include all words i.e. high recall. We use recall in text detection because our aim is not to miss correct word boxes with the cost of introducing false detections.

Our work is similar to the recent works [66, 56] in terms of providing word box proposals. [66] combines two generic object proposal outputs, namely Edge Boxes [201] and Aggregate Channel Feature Detector [27], as preliminary word box proposals. Then, these proposals are filtered using the HOG [22] feature with a Random Forest text/non-text classifier [10]. Finally, the remaining word box proposals are processed using a convolutional neural network regressor to refine the coordinates of these word boxes. [56] performs an over-segmentation using maximally stable extremal region (MSER) algorithm with flexible parameters. Then, the segments are grouped together using distance metrics related to text (e.g. color, stroke width etc.). Finally, weak classifiers are used to obtain a text-likeliness measure for these word candidates. In contrast, our word box generator is uniquely designed to detect text in images without any training involved. Moreover, [66, 56] in the end aim at high f-score word recognition whereas this chapter aims only at high recall. We experimentally verify high recall is more important than high f-score for the applications considered in this chapter. Further, different from [66, 56] which address word recognition, the aim of this chapter is to combine textual and visual cues for better fine-grained recognition and logo retrieval.

Text Recognition. Text recognition approaches can be categorized into two groups: character and word based methods. Character based methods first recognize single characters, then form words [118, 119, 134]. Recent work [2, 53, 67] shows that entire-word recognition performs better than recognizing characters first and then forming words. In this chapter, we follow the state-of-the-art word recognition approach [67] to encode the textual cues.

Textual Cues. Mishra et al. [117] propose to use textual cues for query-by-text image retrieval. Given a query text, the method assigns scores to images based on the presence of the query characters. Additional pairwise spatial constraints between characters are used to refine the ranking. In Chapter 4, we propose to use textual cues in combination with visual cues for fine-grained classification. Bi- and trigrams are computed based on recognized characters in images. These bi- and trigrams are used to encode the textual cues. In contrast, this chapter performs a word-level textual cue encoding. Moreover, the proposed method aims at high recall word detection which leads to combine state-of-the-art text detectors performed in various color spaces.

Fine-grained Classification. Many recent works in fine-grained classification exploit domain specific knowledge. Dogs and birds are composed of a number of semantic parts, such as head, body and tail. [195, 184, 197] use parts for better fine-grained recognition. [195] learns part detectors and localizes the parts to isolate the subtle differences in specific parts. [184] shows the hidden layers of a deep neural network are actually part detectors and uses the filters in the hidden layers to detect specific bird and dog parts. [197] generates multi-scale part proposals and selects useful parts. [46] presents another successful use of domain specific knowledge for bird species recognition. It exploits the fact that birds have rather fixed poses and fits an ellipse to represent the overall shape of a bird. In this work, we exploit the domain specific knowledge for building types classification. In our case, the domain knowledge is the scene text in the building images. We propose a multimodal approach to fine-grained building type classification by fusing the textual and visual cues. A recent paper from *Google* [124] also studies the classification of different business places. [124] only considers visual cues for classification while we show that adding textual cues significantly outperforms methods that only use visual information.

5.3 Word-level Textual Cue Encoding

In order to extract the textual cues from the image, a two-step procedure is followed. In the first step, word box proposals are generated to locate the words in the image. In the second step, the word proposals are used as input to a word recognizer to form the word-level representation.

5.3.1 Word Box Proposals

High recall. When a word in an image is not detected or localized incorrectly, it is not possible to identify it. Our aim is to obtain high recall with the cost of false positives. To this end, the proposed method uses a complementary set of character detection algorithms and color invariant spaces.

Low computational cost. The word box proposal method needs to be efficient especially for large scale scenarios. Further, the number of possible word box candidates (i.e. proposals) should be as low as possible.

Generic. We aim for a generic word proposal method. No need for tuning the method for different alphabets or datasets.

Therefore, we propose an efficient and fully unsupervised bottom-up approach. First, characters are detected by a text-independent approach. Then, these detected characters are filtered based on geometric and appearance properties. Finally, they are grouped to generate word box proposals.

Character Detection

As stated earlier, there exists no single character detection algorithm that is robust against all variations in text style, location and orientation and imaging conditions. Therefore, we propose to compute character candidates using two methods with different strengths, i.e., text saliency (proposed in Chapter 4) and Maximally Stable Extremal Regions(MSERs) [112].

In Chapter 4, a text saliency map is computed using scene background. It is assumed that background pixels are uniformly colored e.g., windows, boards, roads, buildings, fences etc. and that they contrast with text regions. Accordingly, the method uses background homogeneity to form connectivity between background pixels. The method selects initial background seeds and grows these seeds iteratively until all background pixels are covered (detected). Assuming that text regions have strong contrast with the background [9], text regions will remain uncovered by the region growing algorithm. Finally, the background image is subtracted from the original image to obtain a text saliency map, which is further binarized using [42] to obtain character candidates (See Chapter 4 for details).

Text saliency computation does not require any tuning for varying text size, style and orientation, and is robust to image noise. However, due to the information loss caused by the image boundary priors and the binarization, the method may miss characters. To compensate for this, we enable MSER as another character detection algorithm. MSERs define an extremal region as a connected component of which image values remain stable within the boundary and highly contrast against boundary pixels [112]. MSER regions are widely in use for character detection [15, 128]. MSER is suited for character detection because text regions are usually designed to have uniform appearance (color). Further, they usually have high contrast with their surroundings. However, MSER has certain shortcomings for character detection such as detecting characters in blurry

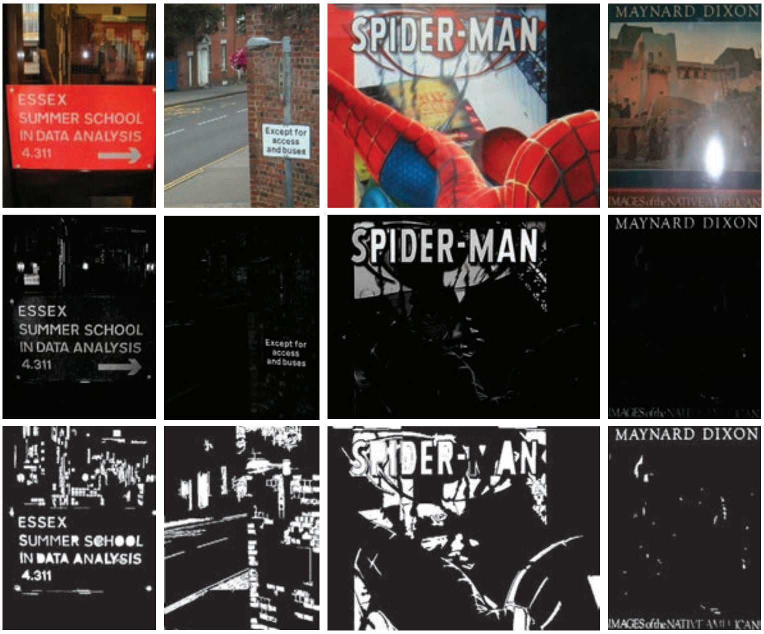


Figure 5.3: Original images (up), text saliency (middle) and MSER character detections (down) obtained. Text saliency method is robust against changes in text size and noise while MSER detects characters at image boundaries.

and noisy images [15]. Moreover, MSER is sensitive to character sizes due to the parameters used to define stable regions. In fact, the MSER and text saliency results are, to a certain extent, complementary. Fig. 5.3 illustrates complementary properties of MSER and saliency methods.

Complementary Color Spaces

Images are captured under uncontrolled illumination conditions. Therefore, text regions may be influenced by different photometric changes such as shadows and specular reflections. A uniformly colored character may vary in intensity due to shadows or highlights. Hence, these shadows or highlights may negatively influence the pixel connectivity for a uniformly colored character.

To compensate for this, the proposed method computes the character candidates using a variety of color spaces containing a range of invariant properties. The two channels, (O_1, O_2), from the opponent color space [35], Saturation (S) and Hue (H) from *HSV* [167], and (I) from gray scale are considered in the proposed method (see Fig. 5.4). The invariant properties are summarized in Table 5.1. Fig. 5.4 illustrates the color channel responses for photometrical changes.

	I	O_1	O_2	S	H
Highlights	-	+	+	-	+
Shadows	-	-	-	+	+

Table 5.1: Color spaces and their invariant properties. I is the gray scale. (O_1, O_2) are the two channels from the opponent color space [35]. Saturation (S) and Hue (H) are from the *HSV* color space [167]. '+' means invariant. In this chapter, we use all these color spaces with different invariant properties to cope with the photometric changes in natural images.

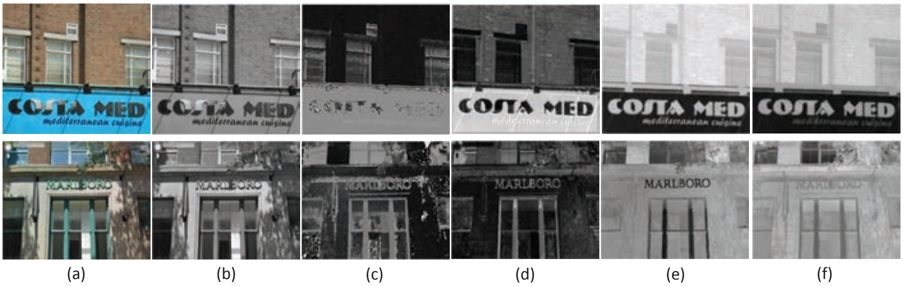


Figure 5.4: Examples of different color channel responses: (a) Original image, (b) Gray-scale, (c) Hue, (d) Saturation, (e) O_1 and (f) O_2. It is shown that color channels have different responses to photometric changes e.g. shadow and highlights, based on their invariant properties.

Character Filtering

The character candidates provided in Section 5.3.1 may consist of non-character regions. Our method for character filtering is based on state-of-the-art text detection systems [15, 31] to filter out non-character regions efficiently.

Aspect ratio. Most of the real characters have a width-height ratio close to 1 [31]. Therefore, the proposed method limits the aspect ratio of character candidates to be a value between 0.1 and 10. These values are reported in [31] to be conservative enough to still keep characters such as 'i', 'I' or '1'. This process filters out text-like items in images such as fences and branches of trees.

Size. The proposed method limits the height of a character candidate to be greater than 5 pixels and the area to contain more than 50 pixels [31]. If the character is too small, the information it carries is limited. Therefore, it is likely that even if these regions are not eliminated, recognition on these regions would fail.

Solidity. The solidity is defined as the proportion of the number of character pixels to the convex area which covers the text candidate. It has been observed that text regions have low solidity [15]. Therefore, the proposed method eliminates character candidates which have high

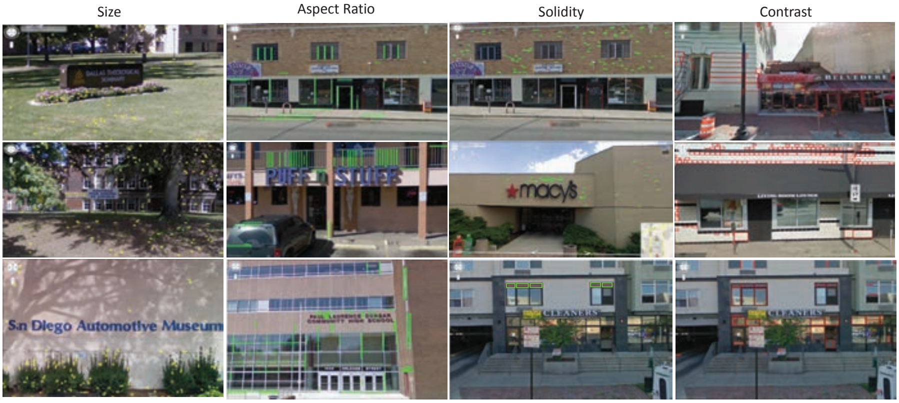


Figure 5.5: Samples for character candidate filtering. The green, red and yellow (different colors are used to highlight boxes) boxes represent filtered character candidates after corresponding filtering condition is applied (i.e. size, aspect ratio, solidity and contrast).

solidity (>0.95) and longer width than height. Longer width is to avoid removing characters like 'i' and 'l'. This process filters out brick-like image regions which have solidity close to 1. Solidity threshold is set to be conservative enough to keep characters like 'w' and 'm'.

Contrast. Pixels at character borders usually have high contrast and the contrast decreases with the distance to the borders. As a result, the box which neatly covers a character will have a higher average contrast than its slightly expanded version. Therefore, the proposed method eliminates the character candidates which do not meet this condition. Contrast (C) of an image pixel (p) is calculated by $C_p = \sqrt{I_x^2(p) + I_y^2(p) + I_x(p)I_y(p)}$, where I_x, I_y are the first order image derivatives (x and y dimensions) in intensity I.

A character candidate satisfying all these conditions is remained for further processing. This filtering step removes those obvious non-character candidates to reduce computational cost in following steps. Fig. 5.5 shows filtered character candidates for each condition.

Word Box Proposal Generation

The next step is to compute word box proposals using character candidates. We consider combinations of character candidates as potential words. However, it is computationally expensive if all possible combinations are considered. And, due to the nature of text, characters within a word cannot have arbitrary positions and sizes [31, 37, 96, 111, 126, 127]. Therefore, as the first step of computing word box proposals, we generate text lines to restrict the selection of combinations by linking character candidates based on five pair-wise constraints. In Fig. 5.6, the two boxes stand for two character candidates with (x_1, y_1), $height_1$ and $width_1$ being the coordinates of the top-left corner, height and width of the box covering the first character. The

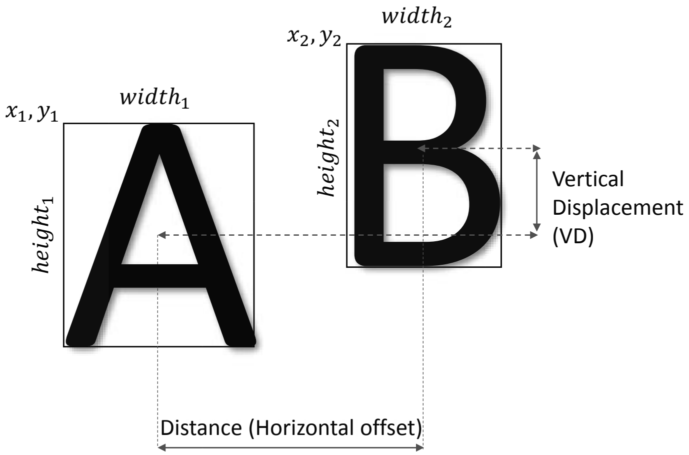


Figure 5.6: An illustration on the notions of two character candidates. This illustration is used to elaborate the pairwise constraints.

box of the second character is defined likewise.

The five pairwise constraints are as follows:

(a) Distance between two character centers is smaller than 2.5 times of the longer axis of the character box [31, 37, 127]. $Distance < \max([height_1, width_1, height_2, width_2]) \times 2.5$ where $Distance = (x_2 + \frac{width_2}{2}) - (x_1 + \frac{width_1}{2})$. 2.5 is considered to allow one missed character in between.

(b) The ratio of the vertical displacement and horizontal offset is no greater than 0.2 [96, 127], formally expressed by $\frac{VD}{Distance} \leq 0.2$ where $VD = |(y_2 + \frac{height_2}{2}) - (y_1 + \frac{height_1}{2})|$ and $Distance$ as defined in (a). Text is mostly horizontally aligned.

(c) The height ratio of two characters is not greater than 2 [31, 37, 127], i.e., $0.5 \leq \frac{height_1}{height_2} \leq 2$. Two characters of a word should have similar height and 2 is considered to allow the case of a lower-case character following a capital.

(d) Two characters must not overlap more than 0.1, formally, $\frac{Area(Char_1 \cap Char_2)}{Area(Char_1 \cup Char_2)} \leq 0.1$. Characters of a word usually do not overlap except in special cases, e.g., italic.

(e) The bottom of one character is below the center of the other [127], i.e., $(y_1 + height_1) \geq \frac{y_2 + height_2}{2}$ and $(y_2 + height_2) \geq \frac{y_1 + height_1}{2}$.Two consecutive characters of a word are usually well aligned for easy reading.

As the second step, we compute word box proposals by considering all possible combinations of character candidates within a text line. A combination of character candidates corresponds to the box covering the union of the character candidates. The proposed method starts with a single character candidate as a word proposal. The reason is that when the characters of a word are connected the word is covered by only one character candidate.

Word box proposals are generated from each character detection algorithm and color space independently and then combined.

5.3.2 Word Recognition and Textual Cue Encoding

Section 5.3.1 generates word box proposals. To recognize words, we employ a state-of-the-art word recognition approach [67]. [67] formulates word recognition as a multi-class classification problem, where a word from a predefined English vocabulary is treated as one class. A convolutional neural network classifier with four convolutional layers and two fully-connected layers is used to solve the classification problem. We refer to [67] for the details of the network. The network takes a word box proposal b as input and produces for each word w a probability of the word being present in the box, $P(w|b)$. The probability is modeled by the softmax scaling of the final multi-way classification layer. As a result, each word box proposal is represented by a n-dimensional feature, where n is the number of words in the vocabulary. In this work, we use the model[†] provided by the authors of [67]. The model considers a vocabulary of $88,172$ words and is trained using synthetic data. We encode the textual cues in an image by summarizing the representations of word box proposals with average pooling. Each dimension of the resulting image feature represents the probability of the corresponding word being present in the image.

5.4 Fine-grained Classification

Fine-grained classification is the problem of the categorization of subordinate-level categories such as bird species [182], flower types [133] and building types [83]. The small inter-class visual differences and the large intra-class variations make fine-grained classification challenging. In this section, in addition to visual features, we exploit the use of textual cues in the images for fine-grained image classification.

5.4.1 Dataset and Implementation Details

Dataset. We use the *Con-Text dataset* proposed in Chapter 4. The dataset is for fine-grained classification of business places e.g., *Cafe*, *Bookstore* and *Pharmacy*. The dataset consists of $24,255$ images from 28 categories. The dataset is divided into three folds. Experiments are repeated three times, each time using two folds as training and the other as testing. We report the mean performance over the three runs. Average precision is used to measure the performance. The first 10 classes of the dataset have word location annotations, whereas the rest of the dataset do not have any available annotation.

[†] http://www.robots.ox.ac.uk/~vgg/research/text

Implementation notes. Three visual-only classification baselines are considered as in Chapter 4. All the three visual baselines employ one-versus-rest SVM classifiers for classification, while the differences lie in the employed visual representations. First, we use a standard bag of visual words representation with 3 × 1 and 2 × 2 spatial pyramid, denoted as *BOW*.

Second, as image representation, we use the $L2$ normalized output of the last average pooling layer of the ImageNet-pretrained GoogLeNet [160], denoted by *DEEP*. The network is pretrained on the 1000 ImageNet categories[‡] [146], available in the Caffe library [70].

Third, we fine-tune the pretrained GoogLeNet with a 28-way softmax classifier on the *Con-Text dataset*. After fine-tuning, the last average pooling layer output of network is used as the image representation. This visual baseline with features from fine-tuned GoogLeNet is denoted by *DEEP-FT*. The details of the fine-tuning are as in Chapter 4. The learning rate is initially set to be 0.001, and is decreased by a factor of 10 every 5 epochs. The network is fine-tuned for 20 epochs. The weight decay parameter equals 0.0005. The network is fine-tuned using SGD with momentum which is set to be 0.9.

For text-based classification, the textual cues are extracted as described in Section 5.3.

Libsvm [14] is used for classification. The histogram intersection kernel is employed for *BOW*, as in Chapter 4, while linear kernel is adopted for *DEEP*, *DEEP-FT* and the proposed textual cues. Textual and visual cues are combined by kernel fusion. Specifically, the visual-based kernel and textual-based kernel matrices are computed independently. Then the two kernel matrices are summed up with equal weights to generate the final kernel matrix. In all experiments, we use the default value for the C parameter (=1) without tuning.

5.4.2 The Influence of Word Detection Precision and Recall on Fine-grained Classification

We use the annotated 10 classes to analyze the effect of word detection precision and recall on fine-grained classification. Therefore, we systematically change recall or precision and evaluate the classification performance. Within this section, only textual cues are used.

Performance on images without text

Not all images in the dataset contain text. However, the proposed method may generate candidate word proposals in non-textual regions in the image. The method uses the character candidate detector using MSER and saliency. Consequently, regions of interest, other than text, may also be detected. We have evaluated the classification performance using the ‘textual cues’ encoded by the proposed method on images without text. Interestingly, it achieves 28.9% in mAP, significantly better than random guessing, although the textual cues are much more effective on images with text (67.7% in mAP). This indicates some salient non-text patterns within the

[‡] `http://www.image-net.org/challenges/LSVRC/2012/`

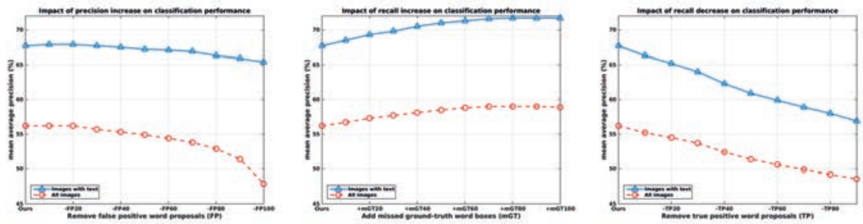


Figure 5.7: The influence of the precision and recall change in word detection on fine-grained classification performance (evaluated on the 10 annotated classes). *Left*: Increasing precision by removing the false positive detections (FP) from the automatically generated set of proposals does not improve the classification performance. *'-FP20'* denotes removing 20% of the false positives. *Middle*: We systematically increase the recall by adding the missed ground-truth word boxes (mGT) on top of the automatically generated set of proposals. The classification performance keeps increasing as the word detection recall increases before it saturates. *'+mGT20'* denotes adding 20% of the missed ground-truth word boxes. *Right*: Decreasing the word detection recall by removing the true positive detections (TP) from the automatically generated set negatively influences the classification performance. *'-TP20'* denotes removing 20% of the true positives. This set of experiments show that word detection recall is more crucial than precision for the classification performance.

same class could be consistently detected and similarly encoded. In the following analysis, we consider two cases, one with images containing text and the other considering all images.

The influence of word detection precision

To study the influence of word detection precision on fine-grained classification, we increase the precision while keeping the recall unchanged by removing the false positive detections (FP) from the generated word proposals. Fig. 5.7 (left) shows that increasing the precision does not improve the classification performance.

Interestingly, this experiment has brought the following additional insight. The classification performance actually decreases when too many false positives are removed from the generated word proposals, especially when all images are considered ('All images'). There are two reasons for this. (1) The proposed word proposal method may detect salient but non-text regions. And some salient non-text patterns within the same class could be consistently detected and similarly encoded. Consequently, some false positive word proposals may contribute positively to the classification, especially for those images without text. This has been discussed in Section 5.4.2. This is also the reason for the decrease in classification performance when removing too many false positives. This is more significant on 'all images' than 'images with text' as shown in Fig. 5.7. (2) The boxes, that cover the text regions for less than 50% overlap with the ground-truth, are treated as false positives. These boxes may contain parts of words or contain complete words with extra background regions. Removing such boxes may have a negative influence on the classification results.

Additionally, we study the influence of precision decrease by adding the generated word proposals (*Ours*) on top of the manually annotated word boxes (*GT*). The classification performance of *GT+Ours* (with a precision of 6.2%) is 75.7% whereas *GT* (with a precision of 100%) is 76.1%. The significant drop in precision from 100% to 6.2% results in a marginal decrease in classification performance.

These experiments indicate that the false positive word proposals generated by the proposed method do not negatively influence fine-grained classification. However, it is worth to mention that it is still desirable to produce a limited number of word proposals for memory and efficiency concerns.

The influence of word detection recall

First, we evaluate the influence of a recall increase on the classification rate. We systematically increase the recall by adding the missed ground-truth word boxes (mGT) on top of the automatically generated set of proposals. As shown in Fig. 5.7 (middle), the classification performance keeps increasing as the word detection recall increases before it saturates.

Second, we decrease the recall by removing the true positive word proposals (TP) from the automatically generated set. The results in Fig. 5.7 (right) show that decreasing the word detection recall negatively influences the classification performance.

Note that even when 90% of the true positive word proposals are removed, the classification performance is acceptable. There are two reasons for this. (1) As discussed in Section 5.4.2 and 5.4.2, the word proposal method is able to consistently detect a number of salient but non-text patterns which are contributing positively to the classification. (2) The boxes that cover the text regions, with less than 50% overlap with the ground-truth, are treated as false positive. Therefore, even when all true positives are removed, these boxes contribute positively to the classification result.

Additionally, we evaluate the performance only using ground-truth boxes. In the case where only images containing text are considered, the performance is 76.1%. When all images are considered, the performance is 54.2%, outperformed by *Ours* (56.2%). When using the ground-truth boxes, the performance on images with no text is random, while when using our generated word proposals, the classification on images with no text is 28.9% in mAP, much better than random guessing (as discussed in Section 5.4.2). This is why when all images are considered, including both images with text and images without text, the performance of using our generated boxes is slightly better than the result of using ground-truth boxes.

Comparison to state-of-the-art text detection

We compare the proposed word detection method with a recent state-of-the-art text detection approach [96]. The textual cue encoding and the classification steps are kept same. [96] aims for

	Performance (mAP%)	
	Ours	*Characterness [96]*
Images with Text	67.7	37.8
All Images	56.2	30.6

Table 5.2: Comparison to state-of-the-art text detection [96]. [96] aims at a high F-score. The recall, precision and F-score values of the proposed method are 64.7%, 4.7% and 8.7% respectively while the values of [96] are 19.3%, 25.3% and 21.9%. A high recall value is more effective than a high f-score for the fine-grained classification problem.

a high f-score, like other state-of-the-art text detection methods [128, 178].

The recall, precision and f-score values of the proposed method are 64.7%, 4.7% and 8.7% respectively while the values of [96] are 19.3%, 25.3% and 21.9%. Compared to [96], the proposed method achieves a significantly higher recall but a lower precision and F-score. In terms of fine-grained classification performance, as shown in Table 5.2, the proposed method (*Ours*) significantly outperforms [96].

5.4.3 Performance evaluation on 28 classes

In this section, we use all 28 classes for evaluation and conduct two experiments. First, we evaluate the effectiveness of the textual cues encoded by the proposed method on the 28-class classification problem. Second, we compare the classification performance of word-level and character-level textual cue encoding.

Experiment I. Three different ways to generate word box proposals are considered: (1) the proposed method using all color channels, denoted by *full*, (2) the proposed method using only the gray scale, denoted by *gray-only*, and (3) a state-of-the-art text detection approach [96] aiming at a high f-score, denoted by *characterness*. We evaluate the sets of word box proposals generated by these three different ways separately while keeping the textual cue encoding and classification steps the same.

As shown in Table 5.3, *full* always outperforms *gray-only* and *characterness* thanks to a higher recall in word detection. The proposed textual-only classification method obtains a mean average precision of 38.3%, outperforming *BOW* (34.0%). The combination of textual and visual cues improves the visual-only baseline by 21.8%, 17.7% and 14.2% for *BOW*, *DEEP* and *DEEP-FT* respectively. It can be derived that recognized words in images contain discriminative information and that it is complementary to visual cues.

Additionally, it is observed that the BOW representation and the deep representation are complementary. Combining BOW, deep representation and the textual cue improves slightly over the combination of deep visual cue and textual cue. Specifically, ‘Textual (full) + Visual (DEEP) + Visual (BOW)’ improves ‘Textual (full) + Visual (DEEP)’ from 71.0% to 72.5%, and ‘Textual

	Performance (mAP%)
Textual-only (full)	**38.3**±**0.9**
Textual-only (gray-only)	33.1±0.5
Textual-only (characterness [96])	20.2±0.6
Visual-only (BOW)	34.0±0.3
Visual-only (DEEP)	53.3±0.08
Visual-only (DEEP-FT)	**60.3**±**0.2**
Textual (full) + Visual (BOW)	55.8±1.0
Textual (gray-only) + Visual (BOW)	52.0±0.6
Textual ([96]) + Visual (BOW)	42.7±0.4
Textual (full) + Visual (DEEP)	71.0±0.5
Textual (gray-only) + Visual (DEEP)	68.7±0.3
Textual ([96]) + Visual (DEEP)	62.0±0.2
Textual (full) + Visual (DEEP-FT)	**74.5**±**0.8**
Textual (gray-only) + Visual (DEEP-FT)	72.7±0.5
Textual ([96]) + Visual (DEEP-FT)	67.5±0.6

Table 5.3: Fine-grained classification performance on *Con-Text dataset*. The textual cue encoded by the proposed method is effective. It is complementary to the visual information. *Textual-only (full)*, *Textual-only (gray-only)* and *Textual-only (characterness [96])* only differ in word detection. Textual cue encoding and classification steps are kept the same. *full* outperforms *gray-only* and *characterness* [96] thanks to a higher recall in word detection.

(full) + Visual (DEEP-FT) + Visual (BOW)' improves 'Textual (full) + Visual (DEEP-FT)' from 74.5% to 75.5%.

Fig. 5.8 shows the per-class performance. The low performance of textual cues is due to the lack of scene text, e.g., for classes as *Bistro* and *Massage Center*. However, combining visual and textual cues improves visual-only on all classes. The performance improvement is the highest on the classes where visual cues are not sufficient and textual cues are discriminative, e.g., *Pawn Shop*, *Dry Cleaner* and *Steak House*.

Experiment II. We compare our word-level textual cue extraction method with character-level textual cue extraction proposed in Chapter 4. We compare textual-only and textual+visual performances. Table 5.4 summarizes the results. Word-level textual cue improves character-level textual cue by 4.7% in mAP. It shows that representing the textual information at a word level is more effective than at a character level.

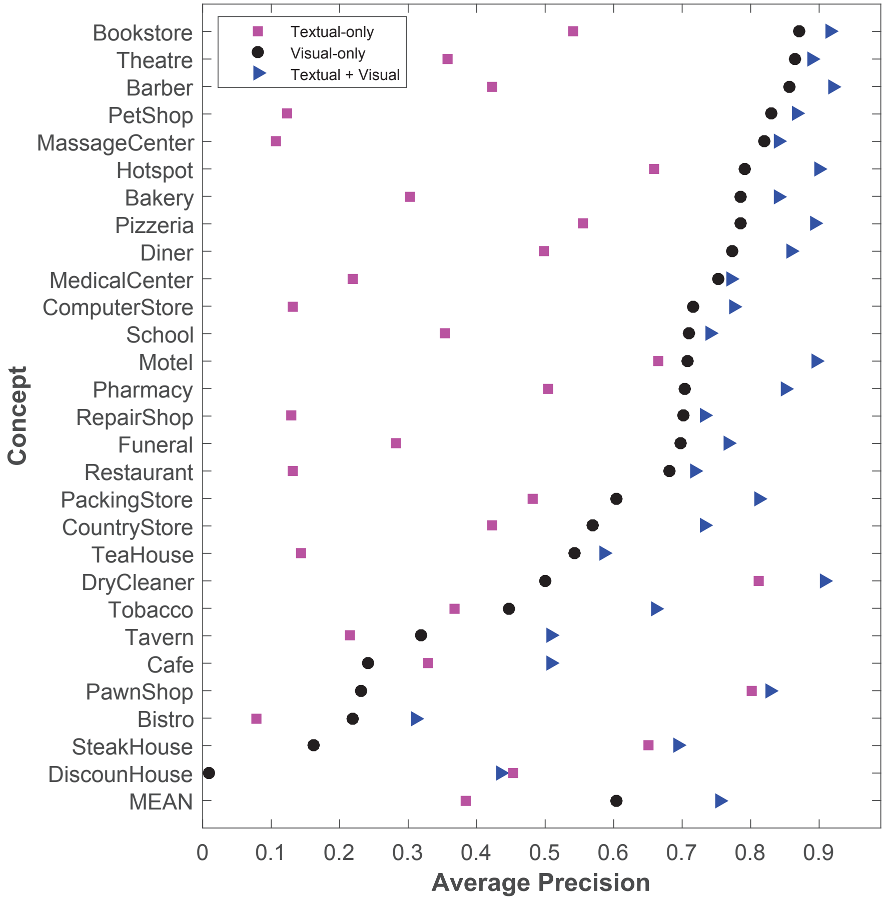


Figure 5.8: Fine-grained classification performance for each class. Adding textual cues improves the performance on all classes. The proposed multimodal approach improves the visual-only baseline (*DEEP-FT*) from 60.3% to 74.5% in mean average precision. Textual-only has average precision values from 10% to 60% and visual-only has values from 0% to 80% largely, whereas multimodal approach guarantees at least 50% except two classes (Bistro and DiscountHouse) up to 90%.

	Performance (mAP%)
Textual-only (Character)	28.4
Textual-only (Word-gray)	33.1
Textual-only (Word-full)	**38.3**
Textual (Character) + Visual (BOW)	47.9
Textual (Word-gray) + Visual (BOW)	52.0
Textual (Word-full) + Visual (BOW)	**55.8**
Textual (Character) + Visual (DEEP)	66.2
Textual (Word-gray) + Visual (DEEP)	68.7
Textual (Word-full) + Visual (DEEP)	**71.0**
Textual (Character) + Visual (DEEP-FT)	70.7
Textual (Word-gray) + Visual (DEEP-FT)	72.7
Textual (Word-full) + Visual (DEEP-FT)	**74.5**

Table 5.4: Word-level textual cues are compared to the textual cues extracted at character-level. Word-level textual-only cue extraction improves our character-level textual-only cue extraction by 4.7% in mAP. It can be derived that representing the textual information at word-level is more effective than at character-level.

5.5 Logo Retrieval

In logo retrieval, the objective is to retrieve all images of a specific logo from an image collection, e.g., *Heineken*, given one image example of that logo as query. Logo retrieval is useful for measuring brand exposure. Logo is a special type of objects where text can be part of the object. Examples are *Starbucks*, *Ford* and *Google*. Previous works [74, 141, 142, 161] do not consider the recognized text of the logo. These methods treat the text of the logos the same as other visual patterns. In contrast, we explicitly extract the word-level textual cues in the logos and utilize it for logo retrieval.

5.5.1 Dataset and Implementation Details

Dataset. We evaluate our approach on *FlickrLogos-32* [143]. *FlickrLogos-32* has 32 brand logos, e.g., *Google*, *Coca-cola* and *DHL*. We follow the retrieval setting of [142], which defines a set of 960 queries, 30 per logo, and a search set of 4280 images in total. The search set consists of 1280 logo images, 40 per logo, and 3000 non-logo images.

Implementation notes. The common method for logo retrieval is to use low level feature matching. In line with this paradigm, two visual baselines are considered. First, we use the available BOW representations with a visual vocabulary of 1 million visual words [142], denoted by *BOW*.

Second, we implement another visual baseline based on aggregated selective match kernels [162], denoted by *ASMK*. The visual vocabulary has 20000 visual words. The kernel we use is a thresholded 4-degree polynomial kernel expressed by $\sigma(\mu) = [\mu > 0]\mu^4$, where the square bracket stands for the Iverson bracket.

For textual cues, we encode the images in the same way as in the previous fine-grained classification application, detailed in Section 5.3. For the query images, we use the query bounding boxes to only keep the word box proposals that overlap with the query boxes. The textual representations are normalized to unit length and cosine similarity is used to rank the images.

To combine the visual and textual cues, we perform a late fusion on the similarity scores obtained from the two modalities. Both sum fusion, expressed by $S_{fusion} = S_{visual} + S_{textual}$, and product fusion, expressed by $S_{fusion} = S_{visual} * (S_{textual} + \epsilon)$ are tested. S_{fusion}, S_{visual} and $S_{textual}$ are the fused score, visual-based score and textual-based score respectively. ϵ is a small constant value added to handle cases where no text has been detected. Sum fusion requires the two scores to be roughly in the same numerical range while product fusion does not. For this reason, only the product fusion is considered for fusing with *ASMK* as the similarity scores produced by *ASMK* lie in a very different range from the scores generated based on the textual cues. The product fusion is also different from the sum fusion because the product fusion has a higher requirement than the sum fusion on the quality of both modalities to derive a decent final result. In general, the product fusion requires both modalities to be reasonably good.

5.5.2 Experiments and Results

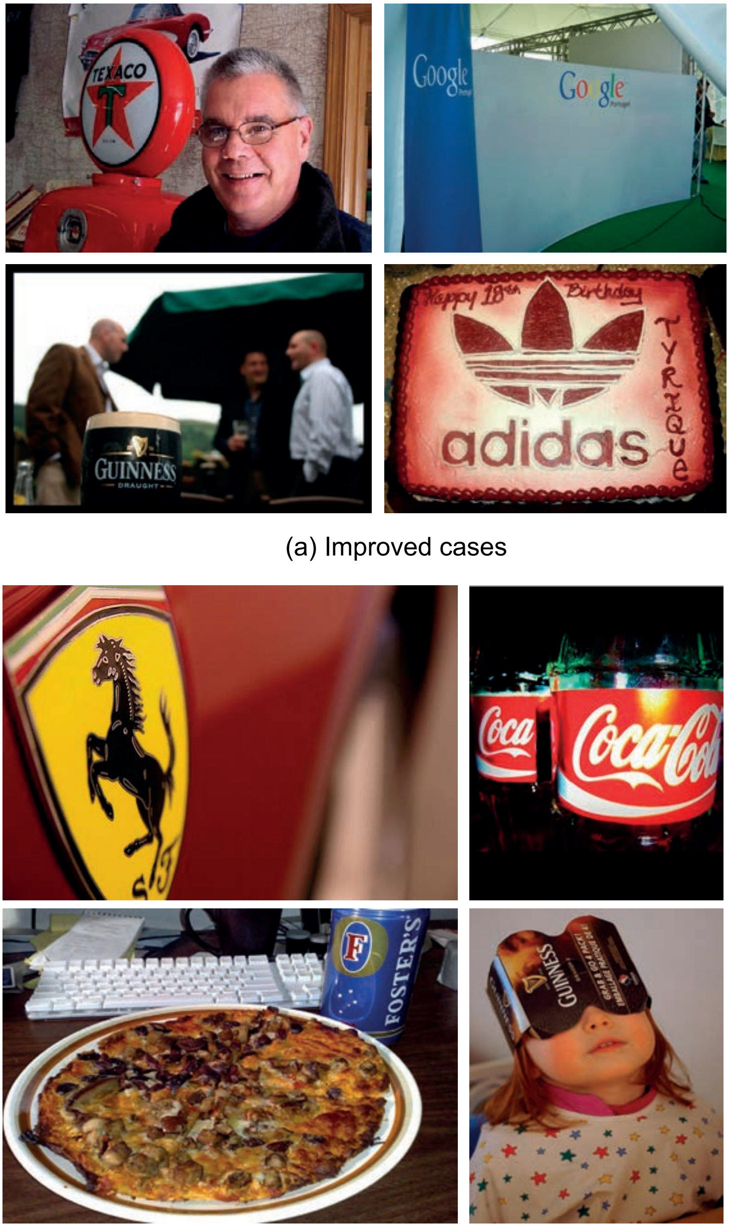
This section experimentally evaluates the proposed multi-modal approach to logo retrieval. We quantify the added value of the proposed textual cues on top of the visual baselines. Moreover, we compare with several state-of-the-art text detection methods for the purpose of logo retrieval.

Table 5.5 summarizes the results. Adding the proposed textual cues 'Textual (full)' and 'Textual (gray-only)' always improves the visual baselines. The best performance, 62.7% in mAP, is achieved by combining the proposed textual cues (full) with the visual baseline (*ASMK*) using product fusion. Interestingly, fusing the textual cues from other text detection methods with the visual baselines using the product fusion does not improve the performance because the performance of the textual part is too modest in these cases. From the experiments, it can be concluded that the proposed textual cue extraction that focuses on high recall word detection is effective, resulting in a textual cue complementary to the visual cues for logo retrieval.

In addition, we compare our word-level textual cue extraction with character-level textual cue extraction proposed in Chapter 4. We compare textual-only and textual+visual performances. The performances in mAP of character-level textual cue extraction are 23.9% and 57.4% for textual-only and textual+visual(BOW) respectively. The performances in mAP of word-level textual cue extraction are 28.4% and 57.8% for textual-only(gray) and textual(gray)+visual(BOW) respectively. Word-level textual cue improves character-level textual cues. It shows that representing the textual information at a word level is more effective than at a character level for logo retrieval.

	mAP%
Textual-only (full)	**32.2**
Textual-only ([96])	12.3
Textual-only (gray-only)	28.4
Textual-only ([192])	13.2
Textual-only ([179])	12.7
Visual-only (BOW)	54.8
Visual-only (ASMK)	**58.4**
Textual (full) + Visual (BOW) [*sum fusion*]	59.4
Textual (gray-only) + Visual (BOW) [*sum fusion*]	57.8
Textual ([96]) + Visual (BOW) [*sum fusion*]	56.0
Textual ([192]) + Visual (BOW) [*sum fusion*]	56.2
Textual ([179]) + Visual (BOW) [*sum fusion*]	55.9
Textual (full) + Visual (BOW) [*product fusion*]	59.5
Textual (gray-only) + Visual (BOW) [*product fusion*]	56.9
Textual ([96]) + Visual (BOW) [*product fusion*]	36.2
Textual ([192]) + Visual (BOW) [*product fusion*]	34.5
Textual ([179]) + Visual (BOW) [*product fusion*]	30.8
Textual (full) + Visual (ASMK) [*product fusion*]	**62.7**
Textual (gray-only) + Visual (ASMK) [*product fusion*]	61.0
Textual ([96]) + Visual (ASMK) [*product fusion*]	41.5
Textual ([192]) + Visual (ASMK) [*product fusion*]	40.1
Textual ([179]) + Visual (ASMK) [*product fusion*]	36.5

Table 5.5: Logo retrieval performance on *FlickrLogos-32* [143]. Adding the proposed textual cues always improves the retrieval performance. The proposed textual cues are more effective than the textual cues from other text detection methods due to the focus on high recall word detection.



(a) Improved cases

(b) Failure cases

Figure 5.9: (a) Example queries where adding textual cues improves the retrieval performance of visual-only. (b) Example queries where adding textual cues decreases the performance. The reasons are no text (*Ferrari*), exotic font style (*Cocacola*) and vertical text (*Foster* and *Guinness*).

Analysis. Adding the textual cues improves the retrieval performance on 641 queries out of 960 ('Textual (full) + Visual (ASMK)'). Text is helpful when it is in standard fonts and orientations. Fig. 5.9(a) shows 4 example queries where combining textual and visual cues improves the performance of visual-only. On the other hand, when text is not there or it is in exotic fonts or orientations, adding textual has a negative effect on the accuracy. Fig. 5.9(b) shows 4 example queries where considering textual information decreases the performance of visual-only. For the query of *Ferrari*, considering textual information is not helpful because there is simply no text. The example of *Cocacola* is due to the exotic font style which makes it unrecognizable. For *Foster* and *Guinness*, the vertical text makes detection and recognition fail.

5.6 Word Box Proposal Evaluation

Dataset. We evaluate the performance of our word box proposal method on the *SVT* dataset [178]. The dataset consists of 249 images which are downloaded from Google Street View of road-side scenes. The dataset has word-level box annotations.

Evaluation measures. The performance is measured in terms of recall, number of proposals and average maximum overlap (*AMO*) (See Table 5.6). We calculate the overlap between each groundtruth box and its best overlapping word box proposal. AMO is the average of these overlap values.

5.6.1 Experiments and Results

We conduct three experiments. First, we evaluate the effect of the color spaces and the character detection algorithms on the word detection performance. Second, we compare our method with state-of-the-art word box proposal methods [66, 56]. Third, we analyze the influence of groundtruth overlap threshold on word detection recall and word recognition accuracy.

Experiment I. The proposed method generates word box proposals using different color spaces and character detection algorithms. Word box proposals are generated for each color space independently and then combined. The same candidate regions may be detected for the different color spaces or character detection algorithms. To filter out these duplicate regions, non-maximum suppression is applied.

Table 5.6 shows that adding more color spaces improves the performance in terms of recall and AMO. When a single character detection algorithm is used, the recall values for MSER and text saliency are 85.47% and 90.88% respectively, whereas the recall is 96.14% when both algorithms are considered. Hence, the use of color spaces with different invariant properties, and complementary character detection algorithms results in a high recall.

Experiment II. We compare the performance of our word proposals with the state-of-the-art word proposal methods [66, 56]. [66] uses generic object proposal methods to generate preliminary word box proposals. However, the number of boxes is prohibitively large ($> 10^4$).

	#proposals	recall(%)	AMO(%)
[This paper] MSER+TSAL, I	338	84.23	70.40
[This paper] MSER+TSAL, I+O_1, O_2+S	806	95.21	77.08
[This paper] MSER+TSAL, I+O_1, O_2+S, H	968	96.14	**77.54**
[This paper] MSER, I+O_1, O_2+H, S	568	85.47	70.90
[This paper] TSAL, I+O_1, O_2+H, S	500	90.88	75.12
TextProposals [56]	17358	94.00	-
Jaderberg et al. [66] without (RF+CNN-reg)	$> 10^4$	**97.00**	77.00
Jaderberg et al. [66] without CNN-reg	900	94.80	-
Jaderberg et al. [66]	900	-	-

Table 5.6: Evaluation of the word box proposals on SVT dataset. *MSER* and *TSAL* are the MSER based and text saliency based character detection algorithms. I, O_1 ,O_2, H and S are the color models. The recall increases as more color invariant models are combined because of their complementary photometric invariant properties. Using both character detection algorithms results in a higher recall than using a single algorithm. RF and CNN-reg of [66] are the Random Forest classifier for non-text box filtering and the convolutional neural network regressor for box refinement. The values for [56, 66] are taken from the references, and empty blocks are not reported in the references. Different from [56, 66] the proposed method is fully unsupervised.

Therefore, [66] filters out most of these boxes using a Random Forest text/non-text classifier. As their recognition step is based on the preciseness of the word boxes, a convolutional neural network regressor is learned to refine the coordinates of the remaining word boxes. [56] uses MSER with flexible parameters and a grouping strategy to generate word proposals. These proposals are also further scored by a weak classifier for word-likeliness. Table 5.6 shows that our method achieves a slightly higher recall than [66, 56] while requiring fewer boxes.

Experiment III. As is common practice in text detection, a candidate word-box is considered as a true positive if it overlaps more than 0.5 with the ground-truth word-box. However, a 0.5 overlap does not guarantee a correct word recognition. In particular, not all true positives are correctly recognized. We analyze the relation between the recognition accuracy and the overlap. The lexicon word with the maximum probability returned by [67] is considered as the word recognition result for each word proposal. Given the word proposals that pass the overlap threshold, the recognition accuracy is computed as the percentage of correctly recognized proposals. Concretely, for a specific overlap threshold, e.g., 0.7, we take all the word proposals that have at least 0.7 overlap with ground-truth, and compute how many of them are correctly recognized. The results are summarized in Fig. 5.10. The results show that the candidate word-boxes (proposals), which have higher overlap with ground-truth, also have higher recognition accuracy. Therefore, not only higher recall but also higher *AMO* is important for accurate textual cue extraction. Further, we vary the ground-truth overlap threshold and evaluate the word detection recall. As expected, increasing the threshold has a negative effect on the recall, see Fig. 5.11. However, the proposed method still performs well for a threshold of (> 0.75). For this threshold

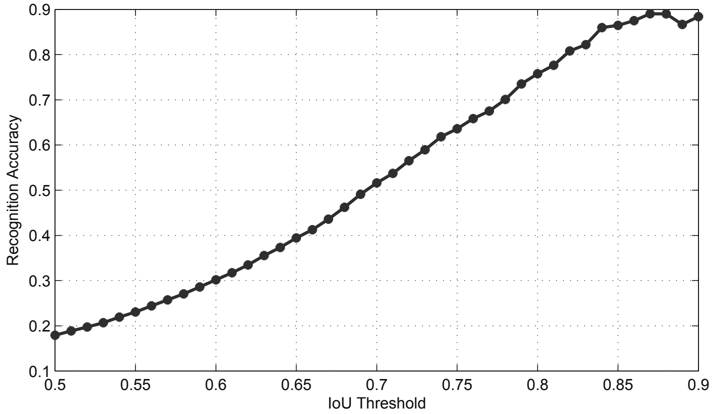


Figure 5.10: The relation between the ground-truth overlap threshold (i.e., the IoU threshold) and the word recognition accuracy, evaluated on *SVT* dataset [178]. Proposals with higher IoU values are better recognizable.

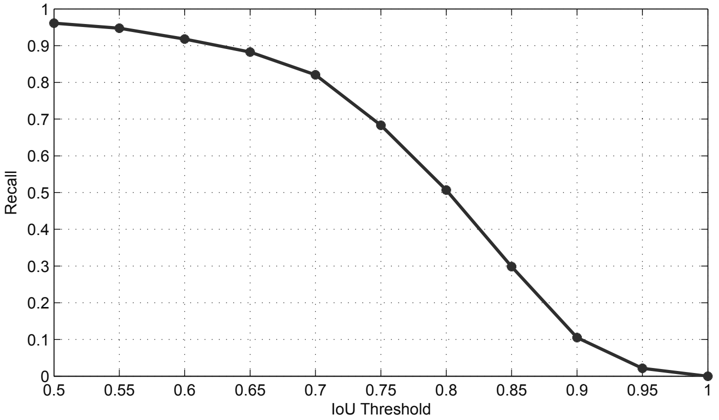


Figure 5.11: The influence of the ground-truth overlap threshold (i.e., the IoU threshold) on word detection recall, evaluated on *SVT* dataset [178]. Recall decreases as IoU threshold increases.

value, the recall and recognition accuracy is around 70%.

In addition, we evaluated the word recognition performance of the proposed method. A common practice to improve word recognition accuracy is to make use of dataset specific dictionaries. However, we did not use the dictionaries provided for this dataset to refine the recognition results. The proposed method reaches a word recognition recall (74.64%) and precision (17.92%).

Efficiency. The matlab implementation of the proposed method (without optimization) takes 4s (on average) on a standard laptop to process one image from the *SVT* dataset.

5.7 Conclusion

We have demonstrated the effectiveness of textual cues for fine-grained (building) classification and logo retrieval. To capture textual information in images, a generic, efficient and fully unsupervised word box proposal approach which aims at high recall has been proposed. For fine-grained building classification, the proposed method outperforms the state-of-the-art [83] from 39.0% to 55.8% in mean average precision. It shows that encoding the textual cues at the word level is superior to using characters. To validate the influence of recall, precision and f-score changes on fine-grained classification, we have annotated a large set of 27601 word boxes. Furthermore, Chapter 5 explores textual cues for logo retrieval. Combining the textual and visual cues improves the retrieval performance to 62.7% from 58.4% of visual-only. Moreover, we show that high recall in word detection is more relevant than high f-score for fine-grained classification and logo retrieval. The proposed unsupervised word box proposal method achieves state-of-the-art recall for word detection on *SVT* with a limited number of word box proposals (< 1000).

6

Combining Object Detectors Using Learning to Rank*

6.1 Introduction

Object detection is an active research area in the field of computer vision. Many detection algorithms have been proposed [41, 173, 86, 110, 52, 151, 167, 19]. Although these detection algorithms are successful for many detection tasks, they may be less accurate for some specific cases.

To gain more insight on the differences amongst detectors, Hoiem et al. [60] provide an extensive analysis on object detectors and their properties [60]. Their findings are that detectors perform well for common object appearances and common imaging conditions. Obviously, different design properties of the detectors (e.g. search strategy, features, and model presentation) influence the robustness of the methods to varying imaging conditions (e.g. occlusion, clutter, unusual views, and object size). For instance, detectors based on the sliding-window approach [41] using pre-defined window sizes and aspect ratios are good at finding likely object positions (rough object positions). However, they are less suited to detecting deformable objects precisely. Hoeim et al. [60] show that these types of detectors typically suffer from poor localization errors. Moreover, the large number of candidate regions to be considered limits the capability of sliding-window based object detection methods [173, 63]. Due to a large number of candidate regions (over $100K$ per image), it is not possible to perform object detection within an affordable time-frame while using strong classifiers [173]. The large number of candidate regions does not only restrict the classifier options but also influences the choice of the selected features. Extracting complex features from a prohibitively large number of sub-regions is not feasible due to its low efficiency [173, 167]. To avoid the limitations of a sliding-window approach, an

*Published as S. Karaoglu, Y. Liu, and T. Gevers, "Detect2Rank: Combining Object Detectors Using Learning to Rank," In the IEEE Transactions on Image Processing (TIP), 25(1):233-248, 2016.

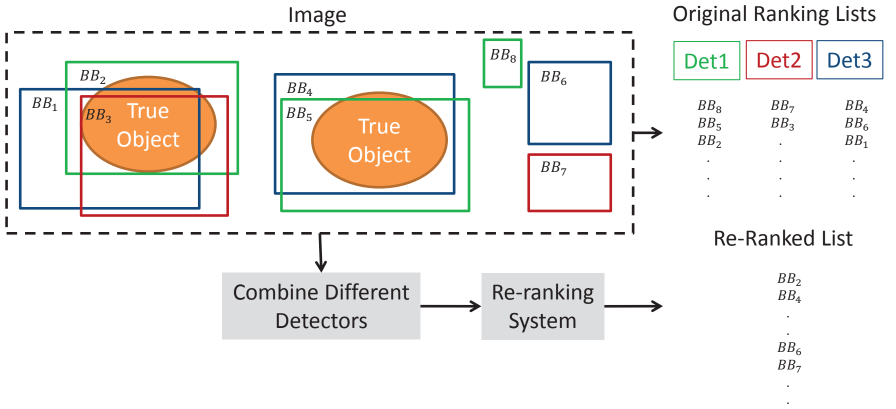


Figure 6.1: Flow of the proposed method (Best viewed in color). Initial detections from different detectors namely, Det1(green), Det2(red) and Det3(Blue) are combined by a learning to rank algorithm. False detections of the individual detectors are learned by detector-detector relations and obtain less confidence when combined, whereas consistency in detectors BB_1, BB_2 and BB_3 are rewarded by the re-ranking system.

object proposal method (selective search [167]), is integrated as a pre-processing step in current state-of-the-art techniques [52]. Selective search generates a significantly reduced set of candidate regions (around $2K$ per image). However, Hosang et al. [63] show that selective search generates candidate regions which are sensitive to changes in scale, illumination and geometrical transformations. This is because selective search is based on segmentation derived from superpixels which are unstable for small image deformations.

Besides the method to generate proper candidate regions for detection, the choice of features influences the robustness and discriminative power of the detectors. HOG-based templates are able to preserve the shape information [41, 110] of objects but are less suited for differentiating between visually similar categories such as cats and dogs. This limitation is addressed using color information in [86], following successful results of using color information in object recognition [168]. HOG-based object detection using color [86] is suited for object classes in which the intra-class color variation is low (e.g. potted plant and tv-monitor). However, the use of color negatively affects the detection accuracy for object classes in which the intra-class color variation is large (e.g. bottles and buses).

Finally, the chosen model and classifier drastically influences the performance of the detectors. In general, object detectors represent all positive samples of a given category as a whole [41, 86]. However, Malisiewicz and Efros [109] show that standard categories (e.g. train, car and bus) do not form coherent visual categories. Accordingly these methods are too generic. To address this issue Malisiewicz et al. [110] propose to train a separate linear SVM classifier for each positive sample in the training set. Gu et al. [57] show that using only one positive sample for training significantly reduces the generalization capacity. Hence, the detection performance of [110] deteriorates for uncommon object views.

As a consequence, no detection algorithm can be considered universal. With the large variety of available methods, the question is how to combine these object detectors to preserve their strengths while reducing their limitations and assumptions. In Chapter 6, we consider a rank learning approach to combine object detection methods. The proposed framework combines detections (detector outputs which consist of a classifier score and bounding box locations) of different well-known object detectors including DPM [41], CN [86] and EES [110]. Furthermore, the method extracts high-level context features such as detector-detector consistency, detector-class preference, object-saliency of a detection, and object-object relations. These features are used in a learning to rank framework to yield a combined detection list. The flow of the proposed method is summarized in Fig. 6.1.

The proposed approach offers the following advantages over single object detectors:

- Missed detections (false negatives) of single detectors are compensated by combining detections of different detectors.
- Detections are re-ranked by using information gathered by other detectors. True detections (true positives) of each detector are rewarded and false detections (false positives) of each detector are penalized within the learning to rank framework.
- The combined list maintains the strengths of the detectors. Therefore, it is more robust than each individual detector for varying imaging conditions.

To the best of our knowledge, we are the first to propose using re-ranking approaches to combine object detectors. Experiments on VOC07 and VOC10 show that the proposed method significantly outperforms single detectors. The proposed method (including code and the detector outputs) will be made publicly available. This allows other researchers to add new detectors.

Our contributions are the following:

- Detector combination: We provide a new perspective on how to approach the object detection problem. As there is no universal object detector, we propose to combine the state-of-the-art object detectors rather than creating a new one.
- Formulating detector combination: We formulate the problem of combining detectors in a learning to rank framework which has not been considered before in object detection.
- Detector contextual integration: We propose high-level context features (e.g. detector-detector relations and object-saliency cues) to combine detections in a learning to rank framework.
- Detector consistency: We show that the state-of-the-art detectors have many detections in common. These common detections are proven to be very informative to re-rank detection scores.

- Detector complementarity: We show that existing state-of-the-art object detectors also have complementary detections. These complementary detections reduce missed detections of single detectors in a combined list.

6.2 Related Work

6.2.1 Object Detection

In general, papers on object detection aim to design a single detector, descriptor or classifier [41, 173, 52, 151, 19, 167, 32]. Felzenszwalb et al. [41] propose a part-based object detection method using HOG features and a latent SVM. This algorithm outperforms the state-of-the-art methods for standard object appearances. The use of template-based models limits a detector's ability to detect deformable objects [60]. Moreover, template-based models (using HOG features) are designed to accommodate for shape information and are less suited to differentiate visually similar categories (e.g cats and dogs). In contrast to part-based detection methods, Vedaldi et al. [173] propose the use of a bag-of-words model for object detection. Multiple features are used within a multiple kernel learning framework which is able to distinguish between visually similar object categories. However, Hoiem et al. [60] show that this approach is sensitive to object size due to the bag-of-words model. Khan et al. [86] propose to use additional color information for object detection. The color information contains expressive power for object classes in which the intra-class color variations are low (e.g. potted-plants or sheep). However, color may have a negative influence on the detection of classes in which the intra-class color variations are high (e.g. bottles or buses) [86].

Malisiewicz et al. [110] propose to learn a linear classifier per exemplar in the training set. The algorithm benefits from a large collection of simpler exemplar classifiers. In this way, the method is tuned to the appearance of the exemplar. While the detections of this detector cover the objects in the dataset (high recall), the detector usually provides low average precision. This is due to the large number of false detections introduced by each of the exemplar specific classifiers. Currently, remarkable results for object detection are obtained by convolutional neural networks [52, 151]. Girshick et al. [52] employ the CNN of [89] to a set of candidate windows obtained by selective search [167]. Recently, Hosang et al. [62] used various object proposals (BING [16], OBJ [1], CORE [139] etc.) to generate candidate windows and evaluate their performance for object detection using RCNN detector. The authors also report the best performance using candidate windows generated by selective search.

6.2.2 Contextual Information for Object Detection

Contextual information for object detection has been exploited over the past few years. Contextual information includes the relation between objects [59, 40], scene layout [23] or characteristics [17, 163], surrounding pixels [59, 12, 43] and background segments [95]. [163] shows that real-world scene structures can be modeled by inference rules. Therefore, in addition to the appearance of objects, contextual information provides useful information for object detection [13, 25]. For example, Choi et al. [17] model the object spatial relationships and co-occurrences by employing a tree-structured graphical model. Desai et al. [23] model the spatial arrangements between objects to detect objects in a structured prediction framework. Cinbis and Sclaroff [18] formulate the object and scene context in terms of relative spatial locations and relative scores between pairs of detections as sets of unordered items. Felzenszwalb et al. [41] re-score their DPM detections by exploiting contextual information as a post processing. Their re-scoring scheme relies on object co-occurrences as well as the location and size of the objects. The above methods show that contextual information is important for object detection. However, these methods have certain limitations. For example, the above methods rely on object-object co-occurrences and spatial relationships and hence are suited for images consisting of (many) different objects. Further, the context-based methods aim at re-scoring detections. They do not introduce new detections and hence are not able to recover from missed detections of single detectors.

6.2.3 Score Aggregation

The approach of aggregating the responses of classifiers and learning a second level SVM to re-score them for different tasks such as action recognition [188], image retrieval [28] and object recognition [164, 158] has been exploited in the literature. The organizers of Pascal VOC12 use seven methods submitted to the classification challenge. The scores of each submission are concatenated to form a single vector to train another linear classifier. Substantial increase for average precision is reported for classes such as potted plants and bottles. However, the problem of aggregating scores of different object detectors is not straightforward as other problems mentioned. More precisely, for these problems each instance in the dataset has a response from each classifier. By contrast, the object detectors do not generate candidate regions (exactly) at the same locations. Therefore, each candidate region does not necessarily contain a response from other detectors. Recently, Xu et al. [185] propose combining different pedestrian detectors through score calibration and detection clustering steps. The authors reduce false and missed detections of pedestrian detectors per image. However, they do not aim to perform a global ranking of detections over the entire dataset for different object classes. In a different work, Ladicky et al. [90] jointly estimates object location and segmentation by minimizing a global energy function on a Conditional Random Field (CRF) model. [90] combines results from detectors (single detector trained for different object classes), pairwise relationships between superpixels, and other low-level cues to perform better segmentation.

6.3 Object Detectors

In this section, the detectors used in Chapter 6 are outlined. We focus on publicly available detectors. Note that there are no constraints on the type of detector since the proposed method only requires detections (bounding box locations with classifier scores) of a detector.

6.3.1 DPM

Felzenszwalb et al. [41] propose an object detector in which each object category consists of a global template and deformable parts. The global template and deformable parts are represented by HOG features extracted at different scales. Training of the object models is done in a latent SVM framework. Each detection $\{x_1, x_2, ..., x_n\}$ in the training set is given a corresponding label, y_i, which is either $+1$ or -1. Each detection x is scored as

$$f_\beta(x) = \max_{z \in Z(x)} \beta.\Phi(x, z). \tag{6.1}$$

The set $Z(x)$ defines all possible latent values for detection x. β and $\Phi(x, z)$ is a vector of model parameters and a feature vector, respectively. β is trained by minimizing the following objective function:

$$L(\beta) = \frac{1}{2}||\beta||^2 + C\sum_{i=1}^{n} max(0, 1 - y_i f_\beta(x_i)), \tag{6.2}$$

where $max(0, 1 - y_i f_\beta(x_i))$ is the hinge loss and constant C is the regularization parameter.

6.3.2 CN

Khan et al. [86] propose an object detector which uses color attributes as an additional feature alongside DPM based HOG features. The color attributes are combined with HOG features in a late fusion manner. The proposed color attributes are compact and efficient. They are proven to be effective for the object classes in which intra-class color variations are low such as potted-plants and sheep. Beside extending HOG features with color attributes, training is done exactly the same as in DPM.

6.3.3 EES

Malisiewicz et al. [110] propose an object detector which is trained by a parametric SVM for each positive exemplar in the training set. Consequently, a large collection of simpler exemplar specific detectors, which are highly tuned to the appearance of the exemplars, are obtained. Each exemplar is represented using a rigid HOG template [21] to train a linear SVM. Then, each Exemplar-SVM, (β_E,b_E), is used as a learned instance-specific HOG weight β_E vector to score. β_E is learned by optimizing the following convex objective function:

$$\Omega_E(\beta, b) = ||\beta_E||^2 + C_1 h(\beta^T x_E + b) + C_2 \sum_{x \in N_E} h(-\beta^T x - b), \tag{6.3}$$

where $h(x) = max(0, 1 - x)$ is the hinge loss and C_1 and C_2 are regularization parameters. Training each detector allows detectors to be tuned based on variations on the exemplar's appearance (viewpoint and object geometry). As a result, high recall is obtained for object detection.

6.4 Combining Detectors by Learning to Rank

To combine detections from different detectors, learning to rank (L2R) is used. L2R aims to rank groups of items according to their relevance to a given task. Fig. 6.2 illustrates a common L2R flow. In our framework, the training set consists of detections $X = \{x_i\}_{i=1}^{m}$ (m is the number of the items in training set) and the ground truth label (y). Feature vector Φ and y are used in training data to learn a ranking model (g). To re-score detections, g is described as follows:

$$g(x) = w\Phi(x). \tag{6.4}$$

Using varied loss functions ξ (see Section 6.5), the weight (w) is optimized by minimizing the following objective function:

$$\min_{w} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i. \tag{6.5}$$

To learn a ranking algorithm that performs re-ranking, the proposed method starts with the feature extraction step using detections x from different detectors.

6.4.1 Context Features

The proposed method starts with high-level context feature extraction to learn how to combine the ranking of detections from different detectors into a single detection list. We aim to extract generic features which exploit the correlation and consistency between detectors.

Detector-Detector Context

We introduce a notion of detector consistency which measures whether different detectors are generating object detections within the same image region. Agreement of all object detectors for a certain location increases the probability of a correct object detection. However, different detectors may generate detections at different locations even for the same image. As a result, it is hard to obtain an exact bounding box location where all detectors provide a detection. Therefore, a relative detector score is defined. To obtain a relative score for each detection, a correspondence term is computed by considering the overlapping ratios between all other detections. In this way, an image is represented as a collection of detections obtained by different object detectors

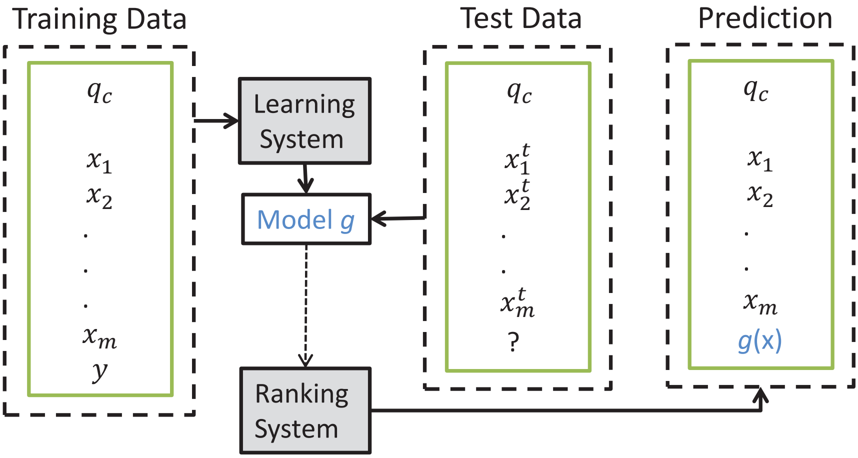


Figure 6.2: Learning to rank framework for detection re-ranking.

j, where $j = \{1, 2 \dots n\}$ and n is the number of the detectors used. For the i^{th} detection in the image, the maximum overlapping detection with each detector is given by:

$$A_{i,j} = \frac{Area(BB_i \cap BB_j)}{Area(BB_i \cup BB_j)} \ , \tag{6.6}$$

$$[\Gamma_i(j), \varphi_i(j)] = \max\left(A_{i,j}\right) \ , \tag{6.7}$$

where Γ is the overlap ratio and φ is the index of the maximum overlapping detection for detector type j. Then, the corresponding relative score R of a detector j to the i^{th} detection is $R_{i,j} = \Gamma_i(j) \times S(\varphi_i(j))$, where S is the initial classification score of the detector. Note that if a detection has no overlap with other detectors ($\Gamma_i(j) = 0$), its relative scores will be zero. In this way, higher relative scores correspond to more reliable detections because more detectors agree on a particular location (see Fig. 6.3). If a detection has high relative score from each single detector it corresponds to a high probability of being a true detection. Whereas a low relative score corresponds to a false detection. Moreover, a mid-level consistency in relative score can be considered as a good indication of poor localization error.

Relative score of a detection does not include the information of which detector it belongs to. However, some detectors perform better than others for some classes, hence their detections should get higher scores than detection of lower-performing detectors (to emphasize the strength of detectors on tasks for which they are successful). Therefore, a detector indicator term is specified. The aim is to provide information to the learning system for identifying detector preferences for particular classes. To give an indication of which detector the detection belongs

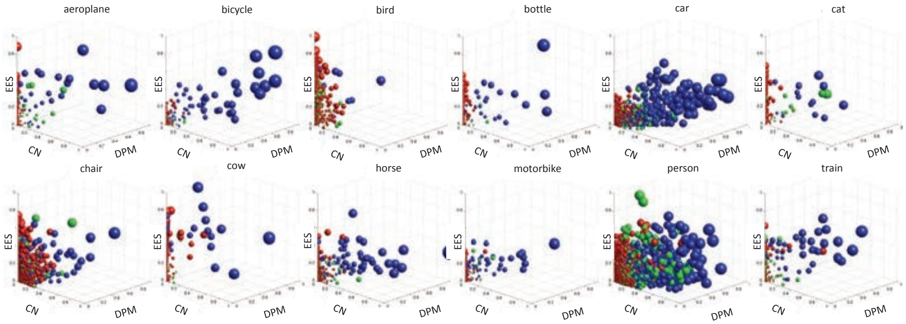


Figure 6.3: The figure illustrates relative score R for each detection in VOC 2007 trainval set. Each sphere represents a detection in the trainval set whereas each axis represents relative score from detectors namely, DPM, CN and EES. The color blue, green and red holds for true detection, poor localization and false detection, respectively. Best viewed in color.

to, a binary vector I_D of three dimensions (i.e. three detectors in our case) is used. The value of the dimension is assigned to be one in case of a detection by the corresponding detector otherwise the value is set to zero. This feature vector is at the detector level. Therefore, all detections of the same detector have the same binary coding I_D.

The final corresponding score feature Rs, for the i^{th} detection is denoted by $Rs_i = \{I_{D,i}, R_{i,1}, R_{i,2}, ..., R_{i,n}, R_{i,1}+R_{i,2}, R_{i,1}+R_{i,3}, R_{i,2}+R_{i,3}, ..., R_{i,n-1}+R_{i,n}, R_{i,1}+R_{i,2}+R_{i,3}+..., +R_{i,n}\}$. The dimension of Rs is limited to the number of the detectors.

Object-Saliency

A feature vector O_s is proposed to represent how likely it is that a detection contains an object. EES [110], OBJ [1] and CORE [139] are used to measure the object-saliency of a detection. OBJ and CORE are category independent region proposal methods. They are mostly used by the current object detection algorithms to avoid an exhaustive sliding window search. These methods provide region candidates/proposals (bounding box) which are likely to contain objects. Both methods result in approximately 1000 candidate regions per image. In addition to these category independent region proposal methods, EES [110] is also used to provide region candidates. The overlap ratios between these different region proposals and object detections are calculated according to eq. 6.6. Then, the feature vector O_s for the i^{th} detection is given by:

$$\Psi_{i,j} = sort(A_{i,j}) \ , \tag{6.8}$$

$$O_s(i,j) = \frac{1}{n}\sum_{k=1}^{n} \Psi_{i,j}(k), \tag{6.9}$$

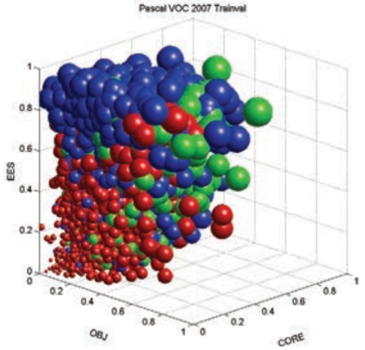


Figure 6.4: The figure illustrates object likelihood score O_s for each detection in VOC 2007 trainval set. Each sphere represents a detection (randomly sub-sampled over all classes) in the trainval set whereas each axis represents object likelihood score from object indicators namely, OBJ, CORE and EES. The color blue, green and red holds for true detection, poor localization and false detection, respectively. Best viewed in color.

where n is the number of neighbors to measure object-saliency, Ψ is the sorted list of overlaps and j is the indicator of different regions proposals, namely OBJ, CORE and EES. Additionally, we use the confidence scores of the maximum overlapping neighbors of detections by EES [110] in eq. 6.9 since these regions proposals are class specific. A detection with a high object-saliency value is considered to be a good indicator for a correct detection. These features may be useful for assigning lower confidence scores to false detections. Fig. 6.4 illustrates that true or false detections are highly correlated with the object likelihood scores.

Object-Object Relation

The likelihood of an object being present is inferred by using other object class likelihoods. Let $S_{c,j}$ be the detection with maximum confidence for object class c ($c = \{1, 2, \ldots, m\}$) by detector j (j={1,2,3}) in an image, where m denotes the number of object classes. Then, the object-object context S_o is given by

$$S_o(c) = \sum_{j=1}^{3} S_{c,j}. \tag{6.10}$$

This feature exploits the object-object relations. For instance, when three detectors locate a cow with high confidence, it is less likely to have a sofa or tv in the same image.

The compactness of the proposed contextual features used in this chapter is shown in Table 6.1.

Feature	Notation	Dimension
Detector Relative Score	R_s	10
Object Likelihood Measure	O_s	4
Object-Object Context	S_o	20
Total		34

Table 6.1: Contextual features used in the proposed learning to rank framework.

We normalize each feature dimension by subtracting its mean and dividing by its standard deviation.

6.4.2 Learning

L2R methods are used to learn the ranking models. L2R methods used in this chapter can be categorized in two groups [101]. The first type of algorithms is called pointwise techniques. Pointwise approaches represent the problem of ranking as a regression or classification problem. These techniques are straightforward approaches to learn the ranking model. Pointwise algorithms are preferred because of their efficiency and effectiveness. These methods have been optimized to work on large scale data.

The second type of L2R algorithms are pairwise techniques. These methods consider the problem of ranking as a pairwise classification problem. The aim is to learn a binary classifier to determine which instance is most relevant from a given pair of instances. The goal of these algorithms is to minimize the average number of misorders in ranking rather than the traditional misclassification in the ordinary pointwise approach.

6.4.3 Non-maximum Suppression

Duplicate removal for the same instance is a known problem for single detectors. Obviously, by combining multiple detectors, the proposed method increases the number of duplicates. To this end, we propose to suppress these multiple detections by non-maximum suppression (nms). The common application of nms considers all bounding boxes (over a certain overlap threshold) for suppression. We use only correspondences (overlaps between detections of other detectors) obtained for each detection in eq. 6.7 for suppression. After applying the re-ranking system, the corresponding detections are sorted and the highest among the others remains constant while detections which are at least 40% covered by the highest detection are suppressed.

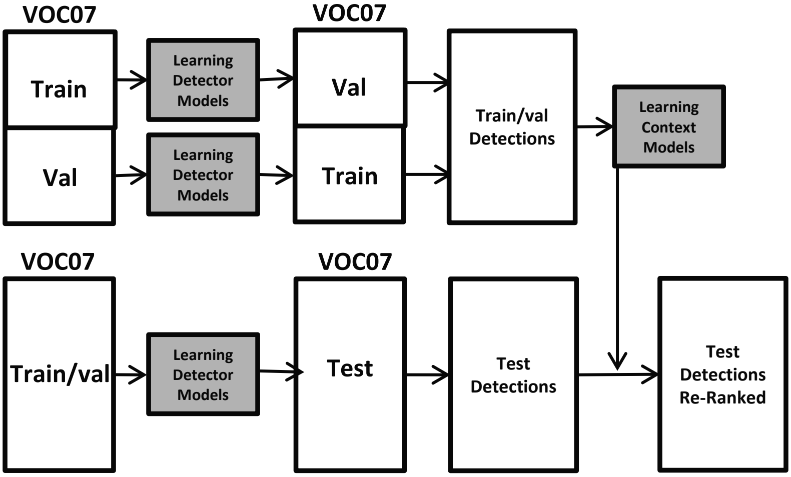


Figure 6.5: Each training is used for learning detector models and context models. To avoid overfitting, the object detectors for context models are trained on the train set to generate detections on validation. Further, they are trained on validation to provide detections on train.

6.5 Experiments

Experiments are conducted on the Pascal VOC07 and VOC10 datasets. VOC07 dataset consists of 9963 images of 20 different object classes (24640 annotated objects) with 5011 training images and 4952 test images. The VOC10 train/val dataset contains 10103 images of 20 different categories (23374 annotated objects). Object detections for the $train$ set are obtained via models trained on $2007val$ and detections for the val set are trained on the $2007train$ set to learn detector-detector context. Detections for the $test$ set are obtained by models trained on the $2007trainval$ set for both dataset evaluations. This process is summarized in Fig. 6.5.

6.5.1 Detector Bounds

In this experiment, we evaluate the maximal mAP that can be achieved by the detections of the baseline detectors and their combinations. The maximal mAP of a detector is calculated when all true detections are ranked at the top of the detection list (precision-recall curve of the maximal AP: precision is always at 1 and the cut-off is at the maximum recall). Since AP corresponds to the area under the precision-recall curve, AP for the maximal AP is $(1 \times max(Recall))$. Consequently, Table 6.2 corresponds to a recall table. Table 6.2 shows that re-ranking DPM, CN and EES detections results in a substantial performance improvement, 17.5%, 16.2% and 33.1%, respectively. This result shows the positive effect of re-ranking detection scores of object

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
DPM [41]	26.7	56.9	2.6	12.8	21.9	46.0	55.3	13.7	19.0	19.4	12.6	2.2	58.1	47.3	40.9	6.8	15.0	26.9	43.4	38.8	28.3
CN [86]	28.7	55.9	6.3	11.6	18.2	44.3	55.5	17.7	18.3	20.5	14.9	4.9	57.3	48.9	41.5	15.0	21.8	28.1	44.1	45.7	30.0
EES [110]	17.9	47.2	2.8	10.6	9.1	39.3	40.3	1.6	6.2	15.3	7.0	1.7	44.0	38.1	13.2	4.6	20.0	11.6	35.9	27.6	19.7
M-DPM	39.3	66.5	29.2	25.5	36.2	58.2	73.4	36.3	53.8	33.6	19.9	22.5	74.7	65.5	62.5	35.0	28.5	37.2	66.0	51.6	45.8
M-CN	43.5	61.7	26.1	20.5	34.5	56.8	72.2	39.4	46.3	33.2	22.8	22.5	73.3	63.1	65.0	38.3	38.8	43.9	62.1	60.1	46.2
M-EES	47.7	72.4	38.3	37.3	46.1	64.3	64.1	45.0	44.4	50.8	44.7	43.1	69.5	63.4	54.9	35.6	47.9	50.2	62.8	73.7	**52.8**
M-(DPM + EES)	60.7	80.4	48.6	46.0	54.6	73.7	80.2	59.5	67.6	58.2	51.9	51.3	82.2	73.8	73.1	49.0	51.7	60.3	76.2	76.0	63.7
M-(DPM + CN)	48.8	68.0	36.8	27.4	40.7	62.9	77.4	49.4	61.5	39.8	31.6	33.7	78.7	70.8	71.2	48.5	42.1	49.4	70.6	61.7	53.5
M-(EES + CN)	59.3	79.5	46.6	43.7	54.6	72.8	78.9	62.0	63.2	55.7	51.5	52.1	82.5	71.7	74.7	50.0	55.8	66.1	73.4	76.3	63.5
M-All	62.5	81.3	52.3	47.5	56.7	76.1	82.3	65.9	71.2	59.0	55.3	56.9	84.2	75.7	77.5	56.0	57.0	67.4	78.4	76.3	**67.0**

Table 6.2: mAP values for baseline detectors DPM, CN and EES. Class specific and overall maximal mAP values of baseline detectors M-DPM, M-CN and M-EES, and their combinations M-(DPM+CN), M-(CN+EES), M-(DPM+EES) and M-(All) on PASCAL VOC07.

detectors.

Table 6.2 shows that DPM and CN have similar maximal mAPs of 45.6% and 46.2%, respectively. However, their combination has a significantly higher maximal mAP (53.5%) than both of them individually. This shows that although these two detectors are very similar in nature, they have complementary detections. Furthermore, when the detectors have intrinsically different designs (e.g. DPM and EES or CN and EES), they produce more complementary detections. This can be derived by the performance gain obtained by combining DPM+EES and CN+EES in Table 6.2, 10.9% and 10.7%, respectively. Consequently, the proposed method would benefit from more detectors.

Another observation that can be derived from Table 6.2 is that aside from detectors having complementary detections to each other, they also have detections in common. While these shared detections are useful to learn consistency in their output, complementary detections compensate for missed detections from each individual detector.

Table 6.2 shows that the performance of detectors is limited by their correct detections. Therefore, detector combinations always show higher mAP values than individual detectors. The proposed method highly benefits from this, whereas other context based re-ranking methods lead to a limited performance improvement (limited to correct detections of a single detector).

6.5.2 Direct Combination of Detections

In this experiment, several ways of combining (without learning) detector outputs are investigated. Because the detectors are trained independently, detector scores are not necessarily compatible. A calibration process [136] is applied before merging different detector outputs. Given a detection x and the learned sigmoid parameter (α, β), the calibrated detection score is calculated as

$$f(x|\alpha,\beta) = \frac{1}{1+exp(x\alpha+\beta)}, \tag{6.11}$$

where α and β for each detector are learned on the $trainval$ set. After the scores are calibrated, we evaluate three different approaches for combining detections:

- $\mathbf{NaiveI}$, after scores are calibrated, detections are merged into a single list.
- $\mathbf{NaiveII}$, after scores are calibrated, detections are sorted in a descending score order for each single detector. Then, detections are combined by taking one by one from the top of each sorted detector outputs.
- $\mathbf{NaiveIII}$, the detectors are combined based on their training set performance. The output of the best performing detector is first added to the list followed by the others based on their performance.

After the detections are combined in a single list, nms (see Section 6.4.3) is applied. It can be derived from Table 6.3 that naively combining detector outputs outperforms baseline scores. The improvements are due to the increase in recall of the combined detection list.

The minimum performance improvement is obtained by $NaiveII$. $NaiveII$ gives equal importance to each single detector. This means that although EES detections are not precise, they become as important as DPM and CN. Therefore, more false positives are introduced at the top of detection list which negatively affects the detection performance. This result shows the importance of properly weighting the detections.

$NaiveIII$ is expected to perform better than other naive methods since it incorporates the training performances of the baseline detectors. However, the $trainval$ performance of the baseline detectors explains the lower performance of $NaiveIII$. To obtain $trainval$ performance detector models are: a) trained on $train$ to test on val and b) trained on val to test on $train$ (see Fig. 6.5). Since the detectors are trained with fewer samples for $trainval$ detections, baseline performances do not necessarily correspond to their $test$ performances. Training with fewer examples has also an influence on our context models.

6.5.3 Learning to Rank Detectors

In this experiment, four different L2R algorithms are evaluated. The pointwise methods we use are the $L2$-regularized support vector classifier ($PoW1$), the logistic regressor ($PoW2$) and the support vector regressor ($PoW3$). The pairwise method is $RankSVM$ [73] ($PaW1$), since it is commonly used as a pairwise L2R method. Pointwise approaches represent the problem of ranking as a regression ($PoW3$) or classification ($PoW1, PoW2$). It takes as input the feature vectors for individual samples and learns a mapping to the ground truth labels whereas pairwise approach takes as input pairs of feature vectors and maps them into binary labels indicating whether two samples are presented in correct order or not.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
DPM [41]	26.7	56.9	2.6	12.8	21.9	46.0	55.3	13.7	19.0	19.4	12.6	2.2	58.1	47.3	40.9	6.8	15.0	26.9	43.4	38.8	28.3
CN [86]	28.7	55.9	6.3	11.6	18.2	44.3	55.5	17.7	18.3	20.5	14.9	4.9	57.3	48.9	41.5	15.0	21.8	28.1	44.1	45.7	30.0
EES [110]	17.9	47.2	2.8	10.6	9.1	39.3	40.3	1.6	6.2	15.3	7.0	1.7	44.0	38.1	13.2	4.6	20.0	11.6	35.9	27.6	19.7
NaiveI	31.0	61.6	6.1	13.7	22.7	48.9	58.4	19.6	20.5	22.3	19.3	3.9	63.2	52.1	44.3	14.5	22.7	31.5	47.8	47.4	32.6
NaiveII	30.8	57.7	6.1	14.2	20.2	47.6	55.2	13.3	16.7	22.4	20.0	4.4	61.4	50.2	33.4	11.8	23.4	28.0	46.4	41.6	30.2
NaiveIII	28.3	61.3	2.8	13.3	22.8	48.1	58.7	18.5	19.5	15.3	19.0	1.8	61.7	52.6	41.9	14.8	20.0	29.3	48.9	48.3	31.4
PoW1	36.8	62.7	10.0	18.1	24.3	51.6	59.5	21.2	22.5	25.4	22.4	7.8	64.2	57.3	44.9	18.7	26.7	34.1	54.1	47.8	35.5
PoW2	36.7	62.8	13.3	18.4	27.0	52.3	59.9	24.7	21.9	24.8	25.8	10.6	65.4	55.9	44.7	19.2	21.2	37.5	54.0	46.5	36.2
PoW3	35.6	63.1	9.7	17.0	25.0	51.2	60.0	21.3	22.5	25.1	21.5	8.1	65.0	56.4	43.8	18.2	27.0	33.9	53.5	48.2	35.3
PaW1	34.5	59.4	10.2	16.2	19.8	49.5	54.4	24.6	20.7	19.7	24.0	8.0	61.0	51.5	40.9	16.7	25.9	31.1	48.3	41.5	32.9
Imp	8.1	7.1	6.2	5.6	5.1	6.3	4.5	7.0	3.5	4.9	10.9	5.7	7.4	8.4	3.4	4.2	5.2	9.4	9.9	2.4	6.3

Table 6.3: The results using learning to rank algorithms. Naive: Direct merging methods without learning. Imp: The improvement over maximum baseline detector by maximum learning algorithm.

L2R algorithms differ mainly by their loss functions ($\xi(w; x_i, y_i)$) in eq. 6.5. ξ for $PoW1$, $PoW2$, $PoW3$ and $PaW1$ are $\max(0, 1 - y_i w^T x_i)$, $\log(1 + e^{-y_i w^T x_i})$, $(\max(0, |y_i - w^T x_i| - \epsilon))^2$ and $\max(0, 1 + w^T x_i - w^T x_j)$ respectively. w represents weights, x instances, y corresponding labels and ϵ parameter to specify the sensitiveness of the loss.

$Liblinear$ [39] implementations for pointwise approaches and rankSVM implementation by Joachims [73] are used with default parameter settings. Ground-truth overlap ratios are taken as training labels. Pascal VOC (> 0.5) overlap criteria is used to assign positive and negative labels for $PoW1$ and $PoW2$, while overlap ratios are directly used as training labels for $PoW3$ and $PaW1$. $PaW1$ requires pairwise preferences between samples, and these preferences are created based on their ground-truth overlaps. Since there is no preference between samples for which the ground-truth overlap equals 0, we do not generate preferences between those samples.

Table 6.3 shows that the proposed learning to rank approach outperforms the baseline detectors for all classes, DPM(7.8%), CN(6.2%) and EES(16.5%). While learning based methods always perform better, logistic regression ($PoW2$) based learning method performs slightly better than other L2R algorithms. Slightly better performance of classification- over regression-based pointwise methods can be explained by the fact that regressor methods try to predict continuous values and do not pay attention to the strict 0.5 overlap boundary of VOC evaluation. Therefore, errors within this range harm regressor results. However, classifier-based methods attempt to minimize these errors. The performance of $RankSVM$ is slightly lower than other L2R methods. This might be due to unbalanced data. The number of negative samples is significantly larger than positive samples. $RankSVM$ treats all the samples equally, therefore some pairs might be overly emphasized within the model.

Considering the low dimensionality of the proposed feature vector, the feature space may not be linearly separable. Therefore, other non-linear kernel options for the classifier could be tested. However, we avoid learning a non-linear SVM due to its long learning time and the need for costly parameter validation. Therefore, we use a feature mapping method proposed by Vedaldi and Zisserman [174]. A 34 dimensional feature vector is mapped to a higher dimensional feature space. The best performing linear classifier in Table 6.3($PoW2$) is applied to this new

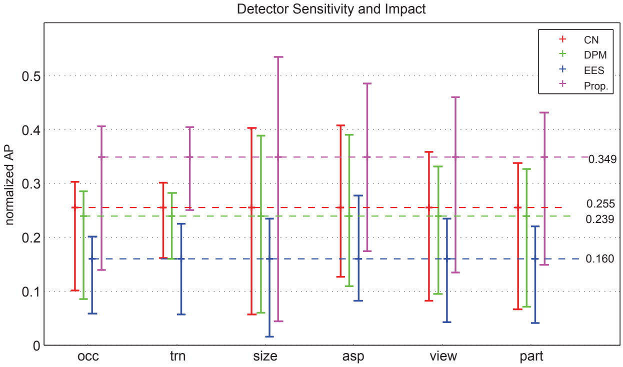


Figure 6.6: Average (over classes) AP_N for the highest and lowest performing subsets within each different object characteristics such as occlusion, truncation, bounding box area, aspect ratio, viewpoint and part visibility.

feature space. Through use of this feature space, the $PoW2$ classifier obtains a 0.6% mAP improvement(36.8%). Increasing the dimensionality results in support vectors which are better able to separate the feature space. Increasing the feature vector dimension with additional context features may further improve the results.

The improvement by the proposed learning scheme over direct merging methods in Table 6.3 indicates that the performance gain is not only due to the increased recall but also the effectiveness of the contextual information and the chosen learning scheme.

6.5.4 Detection Error Analysis

To provide more insight into the performance obtained by combining the baseline detectors, we follow the procedure introduced by Hoiem et al. [60]. Our first analysis regards detector sensitivities. The detector sensitivity is calculated based on the difference between max and min normalized AP for each characteristic (occlusion, truncation, bounding box area, aspect ratio, viewpoint, part visibility). Each colored plot in Fig. 6.6 shows the mean (over all classes) normalized AP for specified detectors. The results show that the proposed method does not reduce the sensitivity. However, it improves both the highest and lowest performing subsets for nearly all object characteristics. This indicates that the proposed method improves robustness for all object characteristics. The sensitivity is not reduced with the proposed method. This is due to commonly missed detections (hard detections cannot be detected easily even for human observers). While some of these hard detections are covered by one of the baseline detectors, they mainly remained undiscovered. That is why the minimum normalized APs for each characteristic increase but not as much as the maximum normalized APs. Consequently, the difference between max and min normalized AP increases.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
R_s	33.7	62.4	9.5	15.2	22.9	50.4	59.6	18.8	22.0	22.8	20.3	6.2	62.7	53.7	44.6	16.5	24.4	34.4	50.9	46.7	33.9
R_s + O_s	35.6	62.1	10.6	17.4	24.6	50.8	59.3	25.1	21.3	23.2	23.9	10.5	63.1	51.0	45.5	14.7	26.3	37.8	50.6	47.5	35.1
R_s + S_o	35.4	63.7	10.6	18.2	26.5	51.7	60.3	18.7	22.7	24.1	21.5	6.6	63.8	57.3	43.7	18.5	24.3	34.5	53.0	45.3	35.0
All	36.8	64.2	12.3	20.3	27.3	53.0	60.3	27.0	22.0	25.3	27.1	11.1	63.7	56.6	45.4	19.3	24.0	38.0	54.5	46.8	36.8

Table 6.4: The influence of selected features for the final detection performance.

Hoiem et al. [60] show the problem of small objects. Since small sized objects are mainly missed by all detectors, we observe that the min normalized AP for category "size" is not improved even if three baseline detectors are combined.

Fig. 6.7 shows the changes in the percentage of each false positive (FP) types with an increasing total number of FP. FPs are divided into four categories as follows:

- Poor localization (Loc) occurs when the label of detection is correct but misaligned with the ground-truth detection ($0.1 \leq$ overlap ≤ 0.5 or a duplicate detection.

- Confusion with similar classes (Sim) occurs when a false detection has an overlap with an instance of a similar class.

- Confusion with dissimilar object categories (Oth) occurs when a false detection is obtained for dissimilar classes.

- Confusion with background (BG) occurs when a false detection has no overlap with an instance of similar or dissimilar classes.

The errors originate from poor localization rather than other errors. This shows the effectiveness of relative score features. For instance, consider an image region where all detectors generate a detection. All detections belonging to this region have high classifier scores because of the high relative score. Consequently, these detections are ranked at the top of the detection list. However, the proposed method creates preferences for certain detectors when dealing with particular classes. Consider a detection by a detector preferred for a particular class that has a localization error within the region. The corresponding detections of the other detectors are suppressed by nms. The suppressed detectors may be true detections. This explains why top ranked false positives of the proposed method are mostly the result of poor localization.

Fig. 6.7 illustrates that the confusion with background error is significantly reduced. This shows the effectiveness of the proposed object likelihood features. Such strong object-saliency cues positively affect the proposed method to detect false detections.

Another observation shown in Fig. 6.7 is that the proposed features could not reduce the confusion caused by similar object categories. However, they are effective on limiting the confusion between dissimilar object categories.

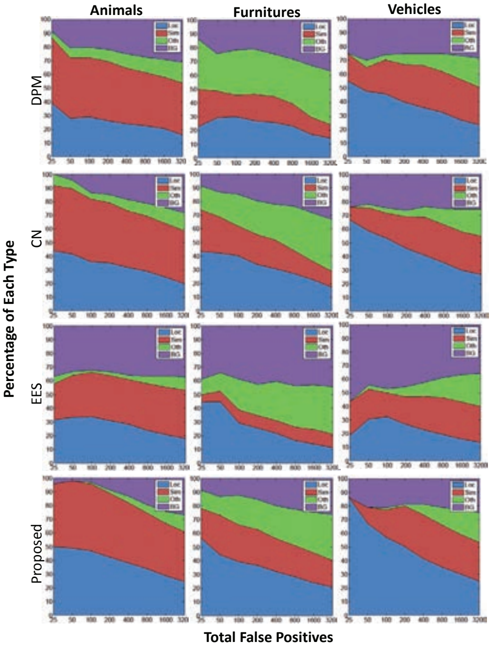


Figure 6.7: Figure shows the fraction of false positives of each type (animal, furniture and vehicle) evolving as the total number of false positives increase.

6.5.5 Feature Importance

In this experiment, we study the influence of each individual feature. The weights are obtained by averaging the absolute classifier weights over the classes. The importance of proposed detector-detector context features (R_s) is highlighted in Fig. 6.8. Moreover, feature weights also emphasize the importance of proposed object-saliency features (O_s). As stated earlier, the proposed R_s and O_s features are more generic and independent of the number of object categories. However, object-object relationships exploited by other state-of-the-art context based object detection methods [17, 23, 18] is dependent on the image characteristics. Therefore, the accuracy gain is limited to the image characteristics for these methods.

We now investigate the influence of each feature on the final mAP score. The detector scores are essential for ranking the detection list. Therefore, it is not possible to evaluate O_s and S_o individually. We evaluate mAP using only the R_s feature. For the rest of the features, R_s is also included. It is shown in Table 6.4 that using only R_s improves the baseline detectors significantly. An object likelihood measure also improves the accuracy (e.g. for animal classes such as cat, dog or sheep). Significant improvement for these classes is due to the poor representation capacity

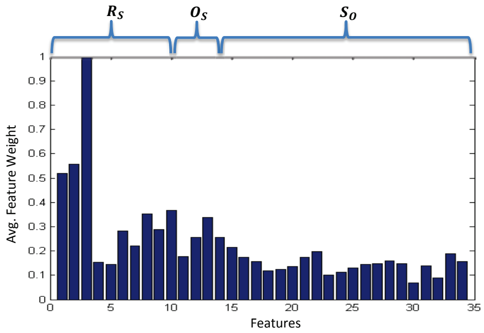


Figure 6.8: Classifier weights are averaged over different classes to see the importance of features individually.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP	Imp.
DPM + Context	29.8	57.5	9.9	16.4	24.1	46.3	58.0	21.1	19.6	20.4	15.1	7.5	58.3	50.4	42.0	14.3	18.2	28.0	49.0	39.6	31.3	3.0
CN + Context	33.3	55.1	11.4	13.4	22.7	44.9	57.0	22.6	18.6	19.4	17.5	8.5	56.0	50.9	42.1	17.4	20.9	31.3	48.5	45.5	31.9	1.9
EES + Context	31.4	57.2	10.6	16.9	21.0	46.6	51.5	13.3	15.5	20.6	15.2	8.1	57.3	51.5	32.9	14.1	18.0	20.1	46.9	44.5	29.7	10.0

Table 6.5: The results of the re-ranked SINGLE baseline detector outputs using contextual features. The results of SINGLE detectors are improved using context.

of template-based detectors for non-rigid objects. Deformable part based object detectors are well suited for detecting rigid parts of the objects (see top ranked visual results of category cat in Fig. 6.10.) Due to the homogeneous appearances of cats, dogs, and sheep, most object proposals contain the full object shape. Therefore, detections of the entire object receive higher confidence than detections for object parts. The object size plays role for other animals, such as horse and cow. Object proposal methods used in this chapter tend to have better performance when detecting small sized objects. Moreover, it is less likely to happen that the object proposal methods generate many large bounding boxes for a specific image region. Therefore, the average overlap of a detection with these windows becomes lower. Adding object-object context (S_o) slightly improves most of the object classes. However, its contribution to the average precision increases when it is combined with the object-saliency. Furthermore, S_o clearly improves the accuracy for class “bottle” in which samples usually occur within a context (usually on a table or in the hand of a person).

6.5.6 Re-ranking Detections from a Single Detector

In this experiment, we exploit the effectiveness of context features without combining detectors into a single list. The proposed context features are only used to re-rank individual detectors. It is shown in Table 6.5 that the proposed method is still effective and improves the baseline detectors. However, the accuracy gain is relatively smaller than using the combined detector outputs in Table 6.3. These results underline the importance of combining different detector

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
BL1	28.6	55.1	0.6	14.5	26.5	39.7	50.1	16.5	16.5	16.8	24.6	5.0	45.2	38.3	35.8	9.0	17.4	22.7	34.0	38.3	26.8
[23]	1.7	0	0.1	1.4	0	-3.5	1.3	0.5	-2.8	1.2	-0.7	0.2	0.5	1.1	-2.8	-1.1	-2.3	-0.7	0.5	0	-0.3
[17]	2.4	-4.2	2.3	0.8	-1.1	-0.2	-0.4	3.9	1.6	0.9	2.3	6.9	5.6	2.2	0	4.7	3.8	2.8	4.7	-0.1	1.9
[18]	5.6	2.7	9.2	0.8	3.2	1.9	3.4	5.0	0	0.7	1.4	7.9	5.9	4.6	3.5	4.2	3.1	4.9	4.9	0.3	3.6
BL2	27.8	55.9	1.4	14.6	25.7	38.1	47.0	15.1	16.3	16.7	22.8	11.1	43.8	37.3	35.2	14.0	16.9	19.3	31.9	37.3	26.4
[200]	2.4	1.9	0.5	0.2	3.2	2.6	2.9	-0.9	0.9	1.9	0.2	5.3	1.3	3.3	3.6	3.0	3.2	3.7	2.9	-0.5	2.0
BL3	26.7	56.9	2.6	12.8	21.9	46.0	55.3	13.7	19.0	19.4	12.6	2.2	58.1	47.3	40.9	6.8	15.0	26.9	43.4	38.8	28.3
Proposed	10.1	7.3	9.7	7.5	5.4	7.0	5.0	13.3	3.0	5.9	14.5	8.9	5.6	9.3	4.5	12.5	8.9	11.0	11.1	8.1	8.4

Table 6.6: Comparison of the state-of-the art context based object detection methods on PASCAL VOC07 dataset. The results of referred works [23, 17, 18] and DPM baseline scores ($BL1$) are reported in [18] whereas [200] and DPM baseline scores ($BL2$) are reported in [200]. $BL3$ is DPM baseline score obtained in this chapter. The results represented as proposed are the improvements over DPM baseline in this chapter.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
DPM [41]	26.7	56.9	2.6	12.8	21.9	46.0	55.3	13.7	19.0	19.4	12.6	2.2	58.1	47.3	40.9	6.8	15.0	26.9	43.4	38.8	28.3
CN [86]	28.7	55.9	6.3	11.6	18.2	44.3	55.5	17.7	18.3	20.5	14.9	4.9	57.3	48.9	41.5	15.0	21.8	28.1	44.1	45.7	30.0
EES [110]	17.9	47.2	2.8	10.6	9.1	39.3	40.3	1.6	6.2	15.3	7.0	1.7	44.0	38.1	13.2	4.6	20.0	11.6	35.9	27.6	19.7
RCNN [52]	62.4	70.9	46.5	37.3	31.8	63.3	72.1	62.3	28.3	64.1	49.2	56.2	66.2	65.2	53.2	28.4	53.1	49.9	57.2	62.2	54.0
Proposed	63.5	74.3	47.1	39.1	38.5	67.1	74.5	62.9	30.7	64.4	50.5	56.3	71.3	68.6	56.4	29.2	53.5	54.2	61.5	63.4	56.4

Table 6.7: The results for baseline detectors CN, DPM, EES, RCNN and proposed detector merging scheme on VOC07 test set. The proposed method outperforms all baseline detectors over all classes.

outputs to recover from missed detections to improve the overall object detection performance.

Note that a detector with a high recall and low precision such as EES can be as powerful as other, more precise detectors (DPM, CN) using the proposed context features.

6.5.7 Comparison to Other Context Methods:

In this experiment, we compare the proposed method against the state-of-the-art context based object detection re-ranking methods. Table 6.6 shows the baseline scores of DPM and improvements reported by the papers [18, 200] on VOC07. The gain in performance by our method indicates the importance of high level contextual features and L2R based detector merging.

Moreover, the proposed method is compared to the recent work by Mottaghi et al. [123] on VOC10 dataset (See Table 6.8). The authors also report on the context re-ranking method of DPM (See [41] for details) discussed in Section 6.2.

The contextual features proposed by other methods in Table 6.6 and Table 6.8 are from different sources. Hence, they can be complementary to the proposed features. Combination of these features may further improve the results.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
BL1	46.3	49.5	4.8	6.4	22.6	53.5	38.7	24.8	14.2	10.5	10.9	12.9	36.4	38.7	42.6	3.6	26.9	22.7	34.2	31.2	26.6
DPM-Context[41]	0.1	1.3	2.7	1.8	-0.6	1.8	2.9	-4.8	0.5	1.3	0.7	1.0	1.5	1.5	2.5	0.6	-2.8	4.9	6.6	2.7	1.2
[123]	6.5	-0.7	7.2	4.4	6.5	1.7	6.9	7.2	0.0	2.1	2.8	3.7	3.4	5.5	2.5	4.6	8.4	3.3	7.9	3.1	4.2
BL2	37.4	51.8	5.1	3.9	20.3	51.4	39.2	13.3	15.2	9.5	7.2	4.8	40.1	43.4	41.5	9.8	13.2	16.4	31.9	26.5	24.1
Proposed	7.3	2.9	8.5	6.6	2.5	7.1	5.1	15.0	2.8	3.5	5.8	7.1	3.6	7.7	3.5	8.0	8.4	3.4	10.6	11.8	6.6

Table 6.8: Comparison of the state-of-the art context based object detection methods on PASCAL VOC10val. The results of referred works [123] and DPM baseline scores ($BL1$) are reported in [123]. $BL2$ is DPM baseline score obtained in this chapter. The results represented as proposed are the improvements over DPM baseline in this chapter.

	aero	bike	bird	boa	bot	bus	car	cat	chr	cow	tab	dog	hor	mbik	pers	plnt	shp	sofa	tra	tv	mAP
[41]	36.8	50.1	4.3	10.6	14.3	50.0	40.4	13.9	15.9	14.2	9.4	4.7	41.8	43.0	40.9	5.9	11.6	15.3	33.4	31.4	24.4
[86]	34.5	48.8	5.3	10.4	11.4	52.1	40.9	18.7	14.9	15.7	7.1	5.9	41.3	45.5	42.2	10.1	14.0	18.1	36.2	35.8	25.4
[110]	22.6	34.9	3.2	9.4	4.5	45.9	25.0	2.1	7.2	10.7	4.3	2.0	21.7	31.7	10.0	2.1	11.6	8.1	21.3	23.6	15.1
PoW2	44.8	53.3	14.3	14.6	14.2	56.3	44.7	27.2	18.9	19.6	14.5	15.0	44.1	50.0	45.4	13.2	17.6	22.5	42.0	39.1	30.6
[41]	37.4	51.8	5.1	3.9	20.3	51.4	39.2	13.3	15.2	9.5	7.2	4.8	40.1	43.4	41.5	9.8	13.2	16.4	31.9	26.5	24.1
[86]	36.6	45.0	6.0	4.7	17.9	52.5	40.2	18.8	15.3	10.6	6.5	5.2	39.7	44.4	44.0	15.5	16.4	13.0	35.6	33.8	25.1
[110]	19.9	36.8	1.8	3.3	7.2	46.2	23.5	2.0	4.2	6.4	2.1	1.3	20.6	30.4	9.5	2.8	14.5	7.0	24.0	24.7	14.4
PoW2	44.7	54.7	13.6	10.5	22.8	58.5	44.3	28.3	18.0	12.9	13.0	11.9	43.7	51.0	45.0	17.8	21.6	19.8	42.5	38.2	30.7

Table 6.9: The results for baselines (DPM, CN and EES) and proposed detector merging scheme using PoW2 on VOC10 (upper: $train$ set and lower: val set).

6.5.8 Increasing Number of Detectors:

We performed another experiment to gain more insight into detector correlations and performance improvement. In this experiment, we focus on the state-of-the-art object detector of [52] (RCNN) in addition to three baseline detectors. The state-of-the-art detector of [52] uses the selective search paradigm [167] to generate object candidates which are classified by convolutional neural networks. [52] obtains the highest detection rate (in the literature) for the Pascal VOC 2007 dataset. The results are summarized in Table 6.7. Table 6.7 shows that the proposed method improves the performance for all classes and outperforms [52]. This indicates that the proposed method is still effective when there is one strong detector (RCNN) which is implemented using substantially different methods than other detectors (CN, DPM and EES). Moreover, Table 6.7 shows that RCNN significantly outperforms other detectors for classes “bird”, “table” and “dog”. However, combining weak detectors (CN, DPM and EES) still provides an improvement in the performance of RCNN for those classes. The maximum gain over RCNN is obtained for classes “bottle” (6.68%) and “horse” (5.12%). Additionally, 16% recall improvement over the single RCNN is obtained. This indicates that CN, DPM and EES still have complementary detections to RCNN.

6.5.9 Tests on VOC10

We also evaluate our method on the PASCAL VOC10 dataset. The VOC10 annotations of the test samples are not publicly available. Therefore, we use only the “$train/val$” dataset. All the training is done on the VOC07 $trainval$ set, including object detection models and detector-detector relation models. Table 6.9 shows the results. Table 6.9 indicates that the proposed method outperforms the baseline detectors for all classes also on the cross dataset evaluation. The results show that the learned detector-detector context is generic and it is not dataset dependent.

6.6 Discussion

6.6.1 Recall and Precision Improvement:

Missed detections of individual detectors are recovered when detections of different detectors are combined in a single list (increased recall). Table 6.2 shows that combining multiple detectors will lead to an increase in true object detections. This indicates that missed detections of individual detectors are recovered by the combined list. Fig. 6.9 also shows that the proposed method only misses objects which are missed by all three baseline detectors.

In [60], it has been shown that detectors can detect objects which contain consistent appearances. They experimentally derive that object detectors have common detections (Section 6.5.1 and Table 6.2). Most detector outputs will overlap for true detections because their aim is to detect the same objects. Therefore, the overlapping information indicates the consistency between detectors and can be used to give more confidence to those detections which overlap with other detections (increase precision). The overlap information is useless for “orthogonal detections”. The question is how to derive more confidence to those “orthogonal detections” to increase their precision. Therefore, the proposed approach makes use of other features such as “I_D-detector indicator”, “object-saliency (O_s)” and “object-object relations (S_o)”. These features are generic and independent of detector orthogonality.

6.6.2 Detector Correlation and Diversity:

In Chapter 6, diversity, and thus potential complementary detections of CN, DPM and EES exist mainly due to three reasons. First, DPM and CN represent all positive samples of a given category as a whole (learn models per-category). However, EES proposes to train a separate linear SVM classifier for each positive sample in the training set (learns models per-sample). Accordingly, DPM and CN are more generic and EES is more discriminative. Second, not only the type of the feature but also how the features are used is crucial for object detection. DPM and CN represents objects using HOG features extracted from object parts and the whole object whereas EES represents objects using HOG features extracted only from the whole object. Moreover, CN uses color information as an additional feature. This results in complementary detections due to

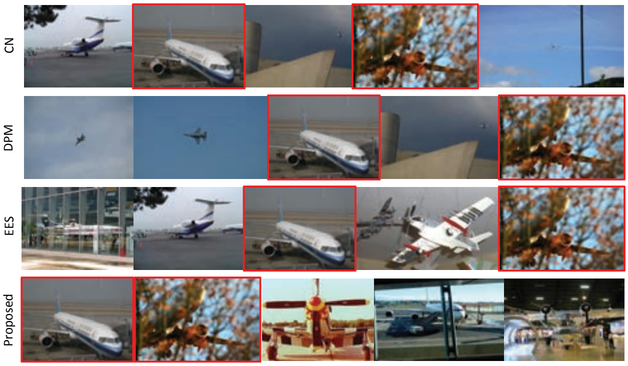


Figure 6.9: The first 5 missed objects by single detectors and proposed method for class "aeroplane". The missed detections are listed based on ascending order of dataset image numbers. The first missed object by "CN" is also missed by "EES", however it is detected by "DPM". Therefore, the proposed method recovers this object. All three baseline detectors missed the objects outlined by red lines. Therefore, the proposed method misses these objects. All missed detections of single detectors which are not outlined by a red line are recovered within the proposed method.

photometric invariance and discriminative power enabled by the color attributes. The discriminative power and photometric invariance do not always guarantee an improved object detection performance. Therefore, CN and DPM detections are complementary to each other. Third, the objective functions are different. DPM and CN minimize the inconsistencies between object parts using latent SVM whereas EES defines per-exemplar distance (more like nearest neighbor search) using linear SVM. These differences have a substantial influence on their final outputs. Table 6.2, Table 6.3 and Fig. 6.12 represent the differences in the final outputs. Although these detectors have detections in common, they have also complementary detections. While common detections are useful to learn consistency in their output to increase precision, complementary detections resolve missed detections for each individual detector.

We also show that when detectors are implemented using substantially different methods, the proposed method still outperforms each individual detector. The experiment conducted in Section 6.5.8 indicates that increasing the number of the detectors will further increase the performance of the proposed method.

6.6.3 Computational Time vs Detection Performance:

There is a tradeoff between computational time and performance improvement. The computational time increases linearly with the number of detectors (assuming the same detection time per detector). The computational time can be reduced either by parallel processing or removing redundant operations e.g. the computation of HOG features, candidate regions etc. With a linear

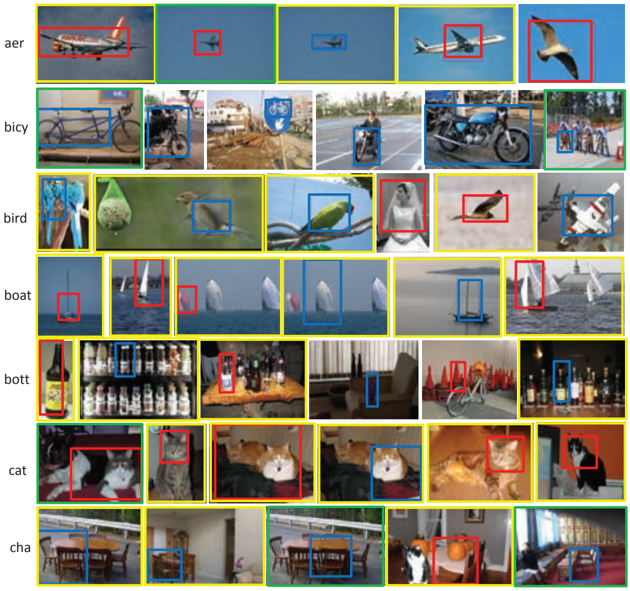


Figure 6.10: Top ranked false positives of the proposed method for specified classes. Blue and red colors indicate the detector type, DPM and CN, respectively. Yellow and green colors correspond to poor localization and multiple detections, respectively. The image frames without color information indicates no overlap between ground truth object, either due to miss classification or background clutter.

increase in time, the improvement in true object detection (recall) starts to slow down eventually. This deceleration depends on the complementarity of the detectors (See Table 6.2). For instance, when CN and DPM are combined, the gain in recall is 7%. When combining EES with CN + DPM, the gain is 13.5%. However, detection performance is not only dependent on the recall. Increasing the number of detectors will yield a higher precision because of an enhanced consistency between detectors. Detections which are supported (overlap) by more detectors are good candidates to be true detections. Hence, agreement between more detectors increases the precision of detections (See Table 6.5). Detections of single detectors become more precise with the help of other detectors. Eventually, the improvement in precision will also slow down (stop) when near optimal detections are provided. Therefore, it can be argued that after a certain detection performance is obtained, the method may stop including more detectors.

6.6.4 The Choice of Learning to Rank Algorithm:

L2R methods are categorized into three groups: pointwise, pairwise and listwise. In general, these three methods differ by their objective functions. All three methods can be used within the proposed framework. However, there are some advantages/disadvantages for each method.

The selection of an algorithm is performed based on the following criteria: scalability, computational complexity and performance. Pointwise techniques are the most straightforward to learn the ranking model. These methods are optimized to deal with large scale data. Hence, they are fast in training and testing. Their main drawback is that the order between samples cannot be considered in the training step. This is because the algorithm optimizes loss functions based on individual sample errors.

A major advantage of the pairwise approach over the pointwise approach is that it focuses on the relative order of samples. Further, like the pointwise approach, the pairwise approach does not consider the position of all the samples in the ranked list. It does not require a quantitative label for each single sample but it requires pairwise preferences between samples. However, in our approach, each detection has a real value (ground truth overlap). Moreover, a pairwise approach is computationally more expensive than the pointwise approach due to the number of pairwise preference constraints formed by the pairs.

The listwise approach considers the position of all the samples in the ranked list when the loss function is optimized. Consequently, the listwise approach has an exponential number of ranking permutations yielding an increase in complexity and computational time. Incorporating the position information in the loss function may further improve the final results. However, the tradeoff between complexity and accuracy should be taken into account.

In fact, accuracy, training speed and memory requirements are important factors to select the number of detectors, object classes and number of training samples. For a (moderate size) dataset such as Pascal VOC (consisting of 20 object classes, 5K samples for training, and three detectors yielding in total 100K detections per class), pairwise solution is still tractable (since most of the detections do not have ground-truth overlap and there is no preference generated between those detections). While it is good to have more training data, it is challenging for pairwise algorithms to handle big data. For a dataset such as ImageNET (consisting of 200 object classes and 500K samples for training) the pairwise approaches need to be further optimized [150] to make the problem tractable. Pairwise approaches are still an area of active research and further improvements are possible by employing recent techniques such as active learning to rank proposed in [137]. Pointwise approaches are already capable of handling such large scale datasets and their performance are proven to be good for this task on Pascal VOC dataset. Therefore, for such large scale experiments pointwise approaches should be used.

6.6.5 Possible Improvements:

To avoid overfitting, the object detectors are trained on $train$ to test on val. Subsequently, they are trained on val to test on $train$, in which case the detectors are trained with fewer examples. This has an impact on the performance of detectors on the $train/val$ set in which we learn the relationship between detectors. It is observed that for some classes the performance of object detectors on the $train/val$ set are not inline with the $test$ set. Therefore, learning the models for detectors on a larger dataset may further improve the proposed learning to rank scheme.

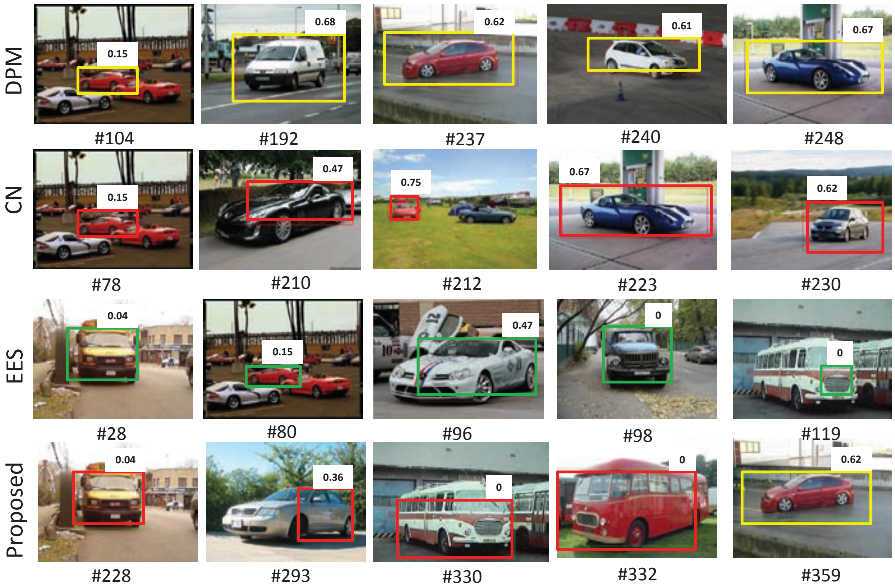


Figure 6.11: Top five ranked false positives of baseline detectors and proposed method with their rankings below (object class car). The color of the detection yellow, green and red indicates the type of detector DPM, CN and EES respectively. The false positives of individual detectors are pushed down in the proposed method.

The non-maximum suppression technique is a widely used ad-hoc method in object detection literature. However, learning to detect multiple detections from different detectors may be more appropriate for the proposed method.

The proposed method does not provide new bounding boxes. Therefore, it cannot recover from poor localization errors. Error resulting from poor localization becomes problematic for some cases (See Fig. 6.10 and Fig. 6.11 for top ranked false positives). This problem can be resolved by proposing new bounding boxes using object proposals or using a method similar to [52].

With the help of the proposed method, future object detectors can focus on more specific solutions to harder detection problems. Their results will be combined with other detection methods to carry object detection algorithms a step further. The contribution of a new method can be compared against the combination of the-state-of-the-art methods.

6.7 Conclusion

No detection algorithm can be considered universal. As a consequence, we have proposed an approach to combine different object detectors. The proposed approach uses (single) object detectors to exploit their correlation by learning a re-ranking scheme.

The proposed method uses the agreement among the detections of different detectors to award a detection based on detector correlation and consistency. Furthermore, the proposed method exploits complementary detections of detectors to help recover missed detections of individual detectors.

Experiments on the PASCAL VOC07 and VOC10 datasets show that the proposed method significantly outperforms individual object detectors (DPM (8.4%), CN (6.8%) and EES (17.0%) on VOC07 and DPM (6.5%), CN (5.5%) and EES (16.2%) on VOC10.)

We show that there are no constraints on the type of the detector. The proposed method outperforms (2.4%) a state-of-the-art object detector (RCNN) on VOC07 when the RCNN is combined with other detectors used in Chapter 6.

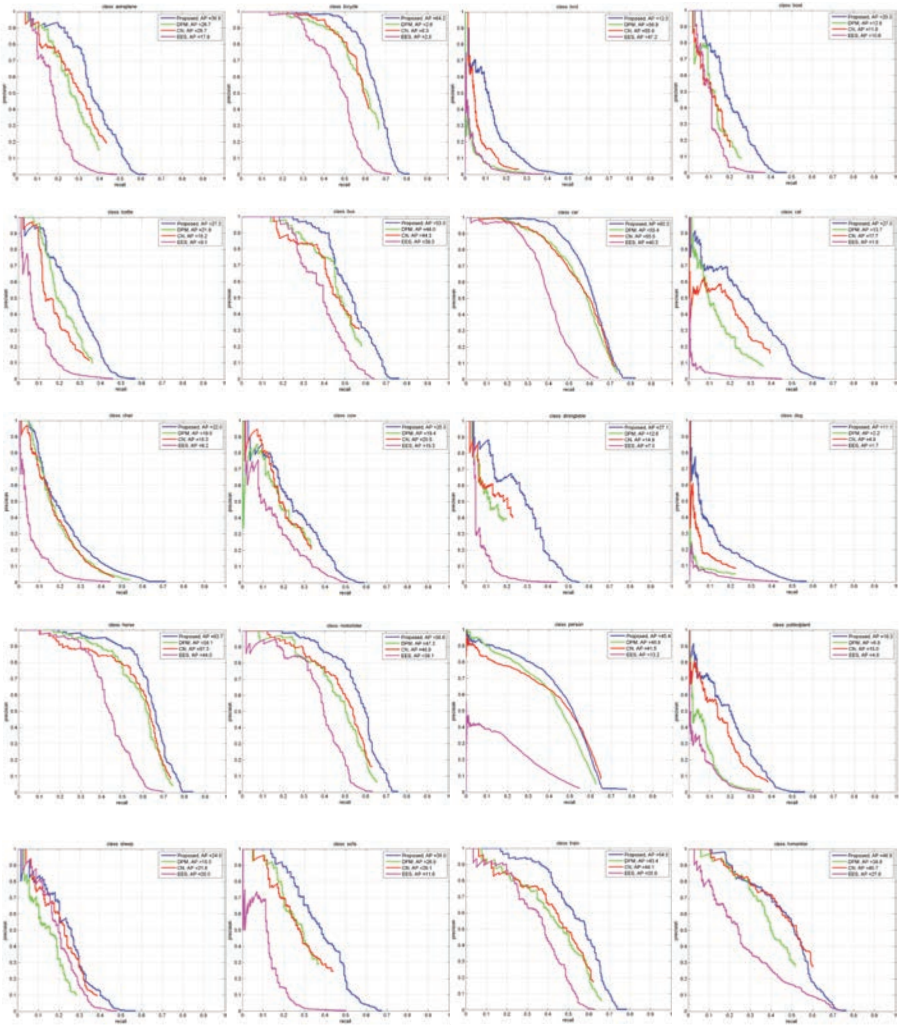


Figure 6.12: Precision-recall curves on PASCAL VOC 2007. The proposed method significantly outperforms all single detectors. Furthermore, it is shown that detections of baseline detectors have remarkable differences.

7

Summary and Conclusions

7.1 Summary

In this thesis, we focus on object recognition and detection for image understanding. We explore images in terms of fundamental elements for what a person needs to see (i.e. light, object and high level semantic interpretation).

Chapter 2: Light Source Position Estimation
In Chapter 2, we focus on detecting the light source position. A common approach for estimating the light source position (LSP) assumes Lambert's law. However, in real-world scenes, Lambert's law does not hold for all different types of surfaces. In Chapter 2, we exploit the influence of surface attributes on the accuracy of LSP estimation. Given a single $RGB-D$ image, we first analyze the effects of photometric (e.g. glossy, matte) and geometric (e.g. curved, smooth) surface attributes. Then the surfaces are ranked using a supervised learning scheme. The ranking results are used to decide the contribution of an image surface segment for LSP estimation. Higher importance is assigned to those image surface segments with suitable photometric (i.e. Lambertian reflectance) and geometric surface attributes. Additionally we assume that light source positions, with respect to static objects in the scene, do not change during a single video recording. Hence, the only change is the relative position of the light source with respect to the camera. Therefore, we propose to use the camera pose to provide temporal constraints in $RGB - D$ sequences. To do this, we first estimate the camera pose to build correspondences between frames. Then, the camera poses are used to optimize the LSP for a $RGB - D$ video sequence.

Experiments on *Boom* and a newly collected $RGB - D$ video datasets show that the state-of-the-art methods are outperformed by the proposed method. The results demonstrate that weighting image surface segments based on their attributes outperforms the state-of-the-art methods in which the image surface segments are considered to equally contribute. In particular, by using the proposed surface weighting, the angular error for light source position estimation is reduced from $12.9°$ to $8.6°$ and $12.9°$ to $7.1°$ for *Boom* and $RGB-D$ video datasets respectively. Moreover, using the camera pose to temporally constrain LSP reduces the angular error $(6.0°)$ compared to using single frames $(7.1°)$.

Chapter 3: Per-Patch Metric Learning for Robust Image Matching
In Chapter 3, we focus on improving matching performance of local image descriptors. Existing methods to extract local image descriptors typically focus on invariance by completely considering, or completely disregarding, all variations. However, a full invariant representation leads to a decrease in discriminative power. In Chapter 3, we propose a metric learning method that is robust to only a range of variations. The ability to choose the level of robustness allows us to fine-tune the trade-off between invariance and discriminative power.

By allowing a degree of invariance, a single global image representation cannot be used as it depends on the specific image content how the limited transformation range will take effect. Therefore, the proposed method is required to achieve robustness on a per-patch basis. We learn a distance metric for each patch independently by sampling from a set of relevant image transformations. These transformations give a-priori knowledge about the behavior of the query patch under the applied transformation in feature space. We present two approaches for learning the distance metric: (i) *full* and (ii) *direct*. The *full* method generates synthetic image patches, extracts descriptors for each patch, and obtains robustness through a metric that is learned on these descriptors. Instead of using the brute-force approach of computing covariance by explicitly generating patches and extracting features from them, in the *direct* method we propose an approach to estimate the covariance from a single patch. We generate a transformation map only once, and use this map to directly estimate the metric from the patch without explicitly generating any synthetic images. The matching between query patch and data is performed with this new metric.

Results on the ALOI dataset show that the proposed method improves performance of SIFT by 6.22% for geometric and 4.43% for photometric transformations.

Chapter 4: Text Detection for Fine-grained Classification and Logo Retrieval
In Chapter 4, we focus on fine-grained object classification using recognized scene text in natural images. While the state-of-the-art relies on visual cues only, we propose to combine textual and visual cues. By incorporating textual cues, we thus add semantics to the recognition process beyond visual cues. The success of the proposed fine-grained object classification method (fusion of visual and text modalities) highly depends on the correctness of the extracted textual image cues. Therefore, a robust character localization and a textual cue encoding method is proposed. Unlike the state-of-the-art text detection methods, we focus more on the background instead of text regions (foreground). Once text regions are detected, they are further processed by two methods to perform text recognition i.e. ABBYY, a commercial OCR engine, and a state-of-the-art character recognition algorithm. Then, to perform textual cue encoding, bi- and trigrams are formed between the recognized characters by considering the proposed spatial pairwise constraints. Finally, extracted visual and textual cues are combined for fine-grained classification.

The proposed method is validated on four publicly available datasets: ICDAR03, ICDAR13, *Con-Text* and *Flickr-logo*. We improve the state-of-the-art end-to-end character recognition by a large margin of 15% on ICDAR03. We show that textual cues are useful in addition to visual cues for fine-grained classification. We show that textual cues are also useful for logo retrieval. Adding textual cues outperforms visual- and textual-only in fine-grained classification (60.3% to 70.7%) and logo retrieval (54.8% to 57.4%).

Chapter 5: Words Matter: Scene Text for Image Classification and Retrieval
In Chapter 5, we exploit word-level textual cues for fine-grained classification and logo retrieval, instead of character-level textual cues extracted in Chapter 4. To detect words in images, a generic and fully unsupervised word box proposal method is introduced. While the state-of-the-art text detection methods aim at high f-score, the proposed method is designed to obtain high recall. A high recall is required because textual cues that are not detected will not be considered in the next (recognition) phase of the framework. Unfortunately, there is no single best method for detecting words with high recall due to large variations in text style, size and orientation. Therefore, we propose to combine character candidates generated by different state-of-the-art detection methods. To obtain robustness against varying imaging conditions, we use color spaces containing photometric invariant properties such as robustness against shadows, highlights and specular reflections. The detected word regions are used as input to a state-of-the-art word recognition method to perform a word-level textual cue encoding.

Results show that adding the textual cues to visual cues improves the mean average precision of visual only in fine-grained classification and logo retrieval, from 60.3% to 74.5% and from 58.4% to 62.7% respectively. Moreover, using word-level textual cues (33.1%) outperforms the fine-grained classification performance of character-level textual cues (28.4%) extracted in Chapter 4. In addition, we show that word detection recall is more important than word detection f-score for fine-grained classification and logo retrieval.

Chapter 6: Combining Object Detectors Using Learning to Rank
In Chapter 6, we focus on detecting objects in images by combining state-of-the-art object detectors. Many object detection algorithms have been proposed. However, each object detector relies on specific assumptions of the object appearance and imaging conditions. As a consequence, no algorithm can be considered universal. In chapter 6, we address the question how to combine the state-of-the-art object detectors. We use detections (detector outputs which consist of a classifier score and bounding box locations) of different well-known object detectors including DPM [41], CN [86] and EES [110]. The method extracts high-level context features such as detector-detector consistency, detector-class preference, object-saliency of a detection, and object-object relations from these object detectors. These features are used in a learning to rank framework to yield a combined detection list.

Experiments on the PASCAL VOC07 and VOC10 datasets show that the proposed method significantly outperforms single object detectors, DPM (8.4%), CN (6.8%) and EES (17.0%) on VOC07 and DPM (6.5%), CN (5.5%) and EES (16.2%) on VOC10. We show with an experiment that there are no constraints on the type of the detector. The proposed method outperforms (2.4%) state-of-the-art object detector (RCNN) on VOC07 when RCNN is combined with other detectors used in Chapter 6.

7.2 Conclusions

In this thesis, we focused on image understanding via object recognition and detection algorithms. Light source is the creator of the image before anything else. Therefore, we have initially approached the problem of light source position (LSP) estimation. Instead of allowing surfaces to equally contribute to LSP estimation, we assigned importance to surfaces based on their photometric and geometric surface attributes. The results have shown that various surface attributes influence the LSP estimation differently. Consequently, learning the importance of surface attributes to estimate LSP outperforms the methods for which image surface segments equally contribute to LSP estimation. In addition, using the camera pose to temporally constrain LSP reduces the angular error compared to using single frames.

Extraction of local image descriptors is important for object recognition. Viewing and lighting condition changes in real-world scenes cause substantial variations in local image descriptor representations. In Chapter 2, we have shown that a full invariant representation leads to a decrease in discriminative power. Instead, we proposed a per-patch metric learning method that is invariant to only a range of variations. We learnt a patch specific a Mahalanobis metric. Two approaches for learning the metric were presented: (i) *full* and (ii) *direct*. The proposed approaches were validated on ALOI dataset for two different image transformations, namely, geometric and photometric. It has been shown that the proposed approaches outperform the original SIFT descriptor matching performance.

Low-level visual cues are not sufficient to understand the deeper meaning of images. It is important to extract higher-level cues as well. Therefore, in Chapter 3, we addressed the problem of object recognition in the scene. Object recognition algorithms have shown significant progress on classifying "distinct" object categories such as horses, bicycles and cars [34]. These algorithms mostly rely on appearance cues (e.g. color, texture and shape) [21, 168]. In Chapter 3, we have shown that these visual cues are not sufficient enough to distinguish categories of objects that only slightly differ in appearance (e.g. fine-grained classification and logo retrieval). Therefore, we have introduced a method to combine textual with visual cues for fine-grained classification and logo retrieval. To extract textual cues, a robust character localization and a textual cue encoding method has been proposed. The proposed algorithm achieves state-of-the-art (end-to-end) character recognition accuracy on the ICDAR03. It has been shown that bimodal information fusion of visual and textual cues increased the fine-grained classification accuracy of visual-only classification by 10.4%. The proposed method outperforms state-of-the-art text detection methods [96] on text saliency and [54, 96, 128] on (end-to-end) character recognition and fine-grained classification. In addition, textual cues proven to be complementary to visual cues for logo retrieval too. Combining textual and visual cues improves the logo retrieval performance over visual-only from 54.8% to 57.4%.

In Chapter 4, the textual cues were extracted at character-level. However, characters cannot fully convey the rich semantics given by the text. To this end, in Chapter 5, we proposed to extract word-level textual cue. Therefore, we proposed a generic, efficient and fully unsupervised word box proposal approach which aims at high recall. For fine-grained building classification, word-level textual cue improves the character-level textual cue from 28.4% to 33.1% in mean average precision. It has been shown that encoding the textual cues at the word-level is superior to using characters. Contrary to what is widely acknowledged in text detection literature, we have experimentally shown that a high recall in word detection is more relevant than a high f-score for fine-grained classification and logo retrieval.

In Chapter 6, we have shown that the design property of object detectors (e.g. search strategy, features, and model presentation) influence the robustness of these methods to varying imaging conditions (e.g. occlusion, clutter, unusual views, and object size). As a consequence, no detection algorithm can be considered as universal. To this end, we have proposed an approach to combine state-of-the-art object detectors. The proposed approach exploited single object detectors and their correlation by learning a re-ranking scheme. We have shown that the shared (common) detections between object detectors, are useful to learn consistency in their output whereas complementary detections compensate for missed detections from each individual detector. Moreover, it is also shown that the performance of detectors is limited by their correct detections. Therefore, detector combinations helped to improve individual detector performances. Experiments on the PASCAL VOC07 and VOC10 datasets have shown that the proposed method significantly outperformed individual object detectors (DPM (8.4%), CN (6.8%) and EES (17.0%) on VOC07 and DPM (6.5%), CN (5.5%) and EES (16.2%) on VOC10.) We have also shown that there are no constraints on the type of the detector. The proposed method outperformed (2.4%) a state-of-the-art object detector (RCNN) on VOC07 when the RCNN is combined with other detectors used in this chapter. With the help of the proposed method, future object detectors can focus on more specific solutions to harder detection problems. Their results will be combined with other detection methods to carry object detection algorithms a step further.

This thesis contributes to various aspects of image understanding, from improving matching performance of local image descriptors (low-level) to recognizing and detecting objects in a scene (high-level). We consider each research question addressed in this thesis as a piece of a complex image understanding puzzle. Solving unique challenges in each piece of the puzzle allows us to reveal the overall picture and acquire in depth image understanding. We believe that the proposed methods, benchmarks, insights and pitfalls represent valuable knowledge that can be leveraged in future research in the area of image understanding (i.e. text and object detection).

Bibliography

[1] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *TPAMI*, 2012.

[2] J. Almazán, A. Gordo, A. Fornés, and E. Valveny. Word spotting and recognition with embedded attributes. In *TPAMI*, 2014.

[3] O. Augereau, N. Journet, A. Vialard, and J.-P. Domenger. Improving classification of an industrial document image database by combining visual and textual features. In *Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on*, pages 314–318. IEEE, 2014.

[4] S. Ayache, G. Quénot, and J. Gensel. Classifier fusion for svm-based multimedia semantic indexing. In *Proceedings of the 29th European conference on IR research*, pages 494–504. Springer-Verlag, 2007.

[5] A. O. Bălan, M. J. Black, H. Haussecker, and L. Sigal. Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In *ICCV*, 2007.

[6] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *JMLR*, 2005.

[7] A. Baumberg. Reliable feature matching across widely separated views. In *CVPR*, 2000.

[8] B. Boom, S. Orts-Escolano, X. Ning, S. McDonagh, P. Sandilands, and R. Fisher. Point light source estimation based on scenes recorded by a rgb-d camera. *BMVC*, 2013.

[9] K. L. Bouman, G. Abdollahian, M. Boutin, and E. J. Delp. A low complexity sign detection and text localization method for mobile applications. *Multimedia, IEEE Transactions on*, 13(5):922–934, 2011.

[10] L. Breiman. Random forests. *Machine Learning*, 2001.

[11] H. Cai, K. Mikolajczyk, and J. Matas. Learning linear discriminant projections for dimensionality reduction of image descriptors. *TPAMI*, 2011.

[12] G. Carolina, M. Brian, B. Serge, and R. G. L. Gert. Multi-class object localization by combining local contextual interactions. In *CVPR*, 2010.

[13] G. Carolina and B. Serge. Context based object categorization: A critical survey. *CVIU*, 2010.

[14] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[15] H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod. Robust text detection in natural images with edge-enhanced maximally stable extremal regions. In *ICIP*, 2011.

[16] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *CVPR*, 2014.

[17] M. J. Choi, A. Torralba, and A. S. Willsky. A tree-based context model for object recognition. *TPAMI*, 2012.

[18] R. G. Cinbis and S. Sclaroff. Contextual object detection using set-based classification. In *ECCV*, 2012.

[19] R. G. Cinbis, J. Verbeek, and C. Schmid. Segmentation Driven Object Detection with Fisher Vectors. In *ICCV*, 2013.

[20] A. Clavelli, D. Karatzas, J. Llados, M. Ferraro, and G. Boccignone. Towards modelling an attention-based text localization process. In *Pattern recognition and image analysis*,

pages 296–303. Springer, 2013.

[21] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[22] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[23] C. Desai, D. Ramanan, and C. C. Fowlkes. Discriminative models for multi-class object layout. *IJCV*, 2011.

[24] J. J. DiCarlo, D. Zoccolan, and N. C. Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

[25] S. K. Divvala, D. Hoiem, J. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *CVPR*, 2009.

[26] D. Doermann, E. Rivlin, and I. Weiss. Applying algebraic and differential invariants for logo recognition. *Machine Vision and Applications*, 9(2):73–86, 1996.

[27] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *TPAMI*, 2014.

[28] M. Douze, A. Ramisa, and C. Schmid. Combining attributes and fisher vectors for efficient image retrieval. In *CVPR*, 2011.

[29] R. O. Dror, T. K. Leung, E. H. Adelson, and A. S. Willsky. Statistics of real-world illumination. In *CVPR*, 2001.

[30] L. Elazary and L. Itti. A bayesian model for efficient visual search and recognition. *Vision research*, 50(14):1338–1352, 2010.

[31] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2963–2970. IEEE, 2010.

[32] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, 2014.

[33] B. Erol and J. J. Hull. Semantic classification of business images. In *Electronic Imaging 2006*, pages 60730G–60730G. International Society for Optics and Photonics, 2006.

[34] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2014.

[35] I. Everts, J. C. Van Gemert, and T. Gevers. Per-patch descriptor selection using surface and scene properties. In *Computer Vision–ECCV 2012*, pages 172–186. Springer, 2012.

[36] I. Everts, J. C. van Gemert, T. E. J. Mensink, and T. Gevers. Robustifying descriptor instability using fisher vectors. *TIP*, 2014.

[37] N. Ezaki, M. Bulacu, and L. Schomaker. Text detection from natural scene images: towards a system for visually impaired persons. In *ICPR*, 2004.

[38] A. Faisman and M. S. Langer. How does lighting direction affect shape perception of glossy and matte surfaces? In *Proceedings of the ACM Symposium on Applied Perception*. ACM, 2013.

[39] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *JMLR*, 2008.

[40] A. Farhadi and M. A. Sadeghi. Phrasal recognition. *TPAMI*, 2013.

[41] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection

with discriminatively trained part based models. *TPAMI*, 2010.

[42] B. Fernando, S. Karaoglu, and A. Trémeau. Extreme value theory based text binarization in documents and natural scenes. In *ICMV*, 2010.

[43] M. Fink and P. Perona. Mutual boosting for contextual inference. In *NIPS*. MIT Press, 2004.

[44] C. Galleguillos and S. Belongie. Context based object categorization: A critical survey. *Computer Vision and Image Understanding*, 114(6):712–722, 2010.

[45] D. Gavrila and J. Giebel. Virtual sample generation for template-based shape matching. In *CVPR*, 2001.

[46] E. Gavves, B. Fernando, C. G. Snoek, A. W. Smeulders, and T. Tuytelaars. Fine-grained categorization by alignments. In *ICCV*, 2013.

[47] J. Geusebroek, G. Burghouts, and A. Smeulders. The amsterdam library of object images. *IJCV*, 2005.

[48] J. Geusebroek, R. van den Boomgaard, A. Smeulders, and H. Geerts. Color invariance. *TPAMI*, 2001.

[49] T. Gevers and W. M. A. Smeulders. Color based object recognition. *Pattern recognition*, 1999.

[50] T. Gevers and H. Stokman. Robust histogram construction from color invariants for object recognition. *TPAMI*, 2004.

[51] A. Gijsenij, T. Gevers, and J. Van De Weijer. Improving color constancy by photometric edge weighting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(5):918–929, 2012.

[52] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[53] V. Goel, A. Mishra, K. Alahari, and C. V. Jawahar. Whole is greater than sum of parts: Recognizing scene text words. In *ICDAR*, 2013.

[54] L. Gómez and D. Karatzas. Scene text recognition: No country for old men? In *Computer Vision-ACCV 2014 Workshops*, pages 157–168. Springer, 2014.

[55] L. Gomez and D. Karatzas. Object proposals for text extraction in the wild. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, pages 206–210. IEEE, 2015.

[56] L. Gomez-Bigorda and D. Karatzas. Textproposals: a text-specific selective search algorithm for word spotting in the wild. *arXiv preprint arXiv:1604.02619*, 2016.

[57] C. Gu, P. Arbelaez, Y. Lin, K. Yu, and J. Malik. Multi-component models for object detection. In *ECCV*, 2012.

[58] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[59] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV*, 2008.

[60] D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *ECCV*, 2012.

[61] B. K. P. Horn and M. J. Brooks. Shape from shading. *Cambridge Massachusetts: MIT Press*, 1989.

[62] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *arXiv:1502.05082*, 2015.

[63] J. H. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? In *BMVC*, 2014.

[64] D. Hu, L. Bo, and X. Ren. Toward robust material recognition for everyday objects. In *BMVC*, 2011.

[65] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth. ACM Symposium on User Interface Software and Technology, 2011.

[66] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *arXiv preprint arXiv:1412.1842*, 2014.

[67] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

[68] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1):1–20, 2016.

[69] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *ECCV*, 2014.

[70] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[71] M. Jiang, S. Huang, J. Duan, and Q. Zhao. Salicon: Saliency in context. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 1072–1080. IEEE, 2015.

[72] Z. S. Jiang, S. Rezvankhah, and K. Siddiqi. Project report: Light source estimation using kinect. 2013.

[73] T. Joachims. Optimizing search engines using clickthrough data. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[74] A. Joly and O. Buisson. Logo retrieval with a contrario visual query expansion. In *ACM MM*, 2009.

[75] N. Joshi and D. J. Kriegman. Shape from varying illumination and viewpoint. *ICCV*, 2007.

[76] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *ICCV*, 2009.

[77] S. Karaoglu, I. Everst, J. C. van Gemert, and T. Gevers. Per-patch metric learning for robust image matching. 2015.

[78] S. Karaoglu, Y. Liu, and T. Gevers. Detect2rank: Combining object detectors using learning to rank. *Image Processing, IEEE Transactions on*, 25(1):233–248, 2016.

[79] S. Karaoglu, Y. Liu, T. Gevers, and A. W. M. Smeulders. Point light position estimation from rgb-d image sequences by learning surface attributes.

[80] S. Karaoglu, R. Tao, T. Gevers, and A. W. M. Smeulders. Words matter: Scene text for image classification and retrieval.

[81] S. Karaoglu, R. Tao, J. C. van Gemert, and T. Gevers. Con-text: Text detection for fine-grained classification and logo retrieval.

[82] S. Karaoglu, J. C. van Gemert, and T. Gevers. Object reading: Text recognition for object recognition. In *ECCV Workshops and Demonstrations*, 2012.

[83] S. Karaoglu, J. C. van Gemert, and T. Gevers. Con-text: Text detection using background connectivity for fine-grained object classification. In *ACM MM*, 2013.

[84] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez i Bigorda, S. Robles Mestre, J. Mas, D. Fernandez Mota, J. Almazan Almazan, and L.-P. de las Heras. Icdar 2013 robust reading competition. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 1484–1493. IEEE, 2013.

[85] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning invariant features through topographic filter maps. In *CVPR*, 2009.

[86] F. S. Khan, R. M. Anwer, J. van de Weijer, A. D. Bagdanov, M. Vanrell, and A. M. López. Color attributes for object detection. In *CVPR*, 2012.

[87] H. I. Koo and D. H. Kim. Scene text detection via connected component clustering and nontext filtering. *Image Processing, IEEE Transactions on*, 22(6):2296–2305, 2013.

[88] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *CVPR*, 2012.

[89] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*. 2012.

[90] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where and how many? combining object detectors and crfs. In *ECCV*, 2010.

[91] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Estimating the natural illumination conditions from a single outdoor image. *International Journal of Computer Vision*, 98(2):123–145, 2011.

[92] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

[93] Y. LeCun, F. J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*, 2004.

[94] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1):29–44, 2001.

[95] C. Li, D. Parikh, and T. Chen. Extracting adaptive contextual cues from unlabeled regions. In *ICCV*, 2011.

[96] Y. Li, W. Jia, C. Shen, and A. van den Hengel. Characterness: an indicator of text in the wild. *IEEE transactions on image processing*, 23(4):1666–1677, 2014.

[97] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014*, pages 740–755. Springer, 2014.

[98] T. Lindeberg. Feature detection with automatic scale selection. *IJCV*, 1998.

[99] C. Liu, L. Sharan, E. H. Adelson, and R. Rosenholtz. Exploring features in a bayesian framework for material recognition. In *CVPR*, 2010.

[100] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur. Dog breed classification using part localization. In *ECCV*, 2012.

[101] T.-Y. Liu. *Learning to rank for information retrieval*. Springer Science & Business Media, 2011.

[102] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[103] S. Lu, T. Chen, S. Tian, J.-H. Lim, and C.-L. Tan. Scene text extraction based on edges and

support vector regression. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(2):125–135, 2015.

[104] T. Lu, S. Palaiahnakote, C. L. Tan, and W. Liu. *Video Text Detection*. Springer, 2014.

[105] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. Icdar 2003 robust reading competitions. In *null*, page 682. IEEE, 2003.

[106] C. B. Madsen and B. B. La. Probeless illuminantion estimation for outdoor augmented reality. *INTECH*, 2010.

[107] V. Mahadevan and N. Vasconcelos. Spatiotemporal saliency in dynamic scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(1):171–177, 2010.

[108] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.

[109] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. In *CVPR*, June 2008.

[110] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011.

[111] J. Mao, H. Li, W. Zhou, S. Yan, and Q. Tian. Scale based region growing for scene text detection. In *ACM MM*, 2013.

[112] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, 2002.

[113] J. michael Frahm, K. Koeser, D. Grest, and R. Koch. Markerless augmented reality with light source estimation for direct illumination. In *In Conference on Visual Media Production CVMP*, 2005.

[114] K. Mikolajczyk and J. Matas. Improving descriptors for fast tree matching by optimal linear projection. In *ICCV*, 2007.

[115] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 2004.

[116] S. Milyaev, O. Barinova, T. Novikova, P. Kohli, and V. Lempitsky. Fast and accurate scene text understanding with image binarization and off-the-shelf ocr. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(2):169–182, 2015.

[117] A. Mishra, K. Alahari, and C. Jawahar. Image retrieval using textual cues. In *ICCV*, 2013.

[118] A. Mishra, K. Alahari, and C. V. Jawahar. Scene text recognition using higher order language priors. In *BMVC*, 2012.

[119] A. Mishra, K. Alahari, and C. V. Jawahar. Top-down and bottom-up cues for scene text recognition. In *CVPR*, 2012.

[120] S. J. Mithen. Looking and learning: Upper palaeolithic art and information gathering. *World Archaeology*, 19(3):297–327, 1988.

[121] J. M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM J. Imaging Sciences*, 2009.

[122] I. Motoyoshi, S. Nishida, L. Sharan, and E. H. Adelson. Image statistics and the perception of surface qualities. *Nature*, 2007.

[123] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014.

[124] Y. Movshovitz-Attias, Q. Yu, M. C. Stumpe, V. Shet, S. Arnoud, and L. Yatziv. Ontological supervision for fine grained classification of street view storefronts. In *CVPR*, pages

1693–1702, 2015.

[125] J. A. Nelder and R. Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[126] L. Neumann and J. Matas. A method for text localization and recognition in real-world images. In *ACCV*. 2010.

[127] L. Neumann and J. Matas. Text localization in real-world images using efficiently pruned exhaustive search. In *ICDAR*, 2011.

[128] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *CVPR*, 2012.

[129] N. Neverova, D. Muselet, and A. Trémeau. Lighting estimation in indoor environments from low-quality images. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 380–389, 2012.

[130] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR)*, 2011.

[131] I. Newton. *Opticks, or, a treatise of the reflections, refractions, inflections & colours of light*. Courier Corporation, 1979.

[132] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1447–1454. IEEE, 2006.

[133] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

[134] T. Novikova, O. Barinova, P. Kohli, and V. Lempitsky. Large-lexicon attribute-consistent text recognition in natural images. In *ECCV*, 2012.

[135] Y.-F. Pan, X. Hou, and C.-L. Liu. A hybrid approach to detect and localize texts in natural scene images. *Image Processing, IEEE Transactions on*, 20(3):800–813, 2011.

[136] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, 1999.

[137] B. Qian, X. Wang, N. Cao, Y. Jiang, and I. Davidson. Learning multiple relative attributes with humans in the loop. *TIP*, 23(12), 2014.

[138] E. Rahtu and J. Heikkilä. A simple and efficient saliency detector for background subtraction. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1137–1144. IEEE, 2009.

[139] E. Rahtu, J. Kannala, and M. B. Blaschko. Learning a category independent object detection cascade. In *ICCV*, 2011.

[140] M. Ranzato and G. E. Hinton. Modeling pixel means and covariances using factorized third-order boltzmann machines. In *CVPR*, 2010.

[141] J. Revaud, M. Douze, and C. Schmid. Correlation-based burstiness for logo retrieval. In *ACM MM*, 2012.

[142] S. Romberg and R. Lienhart. Bundle min-hashing for logo recognition. In *ICMR*, 2013.

[143] S. Romberg, L. G. Pueyo, R. Lienhart, and R. Van Zwol. Scalable logo recognition in real-world images. In *ICMR*, 2011.

[144] P. P. Roy, U. Pal, and J. Lladós. Document seal detection using ght and character proximity graphs. *Pattern Recognition*, 44(6):1282–1295, 2011.

[145] M. Rusiñol, V. Frinken, D. Karatzas, A. D. Bagdanov, and J. Lladós. Multimodal page classification in administrative document image streams. *International Journal on Document Analysis and Recognition (IJDAR)*, 17(4):331–341, 2014.

[146] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[147] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *ICRA*, 2011.

[148] I. Sato, Y. Sato, and K. Ikeuchi. Illumination from shadows. *TPAMI*, 2003.

[149] U. Schmidt and S. Roth. Learning rotation-aware features: From invariant priors to equivariant descriptors. In *CVPR*, 2012.

[150] D. Sculley and G. Inc. Large scale learning to rank. In *Workshop on Advances in Ranking in NIPS*, 2009.

[151] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *ICLR*, 2014.

[152] A. Shahab, F. Shafait, A. Dengel, and S. Uchida. How salient is scene text? In *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*, pages 317–321. IEEE, 2012.

[153] L. Sharan, Y. Li, I. Motoyoshi, S. Nishida, and E. H. Adelson. Image statistics for surface reflectance perception. *Journal of the Optical Society of America*, 2007.

[154] P. B. Sharma. Painting: A tool of non-verbal communication. *Language in India*, 13.

[155] C. Siagian and L. Itti. Rapid biologically-inspired scene classification using features shared with visual attention. *TPAMI*, 29(2):300–312, 2007.

[156] D. Simakov, D. Frolova, and R. Basri. Dense shape reconstruction of a moving object under arbitrary, unknown lighting. *ICCV*, page 1202ÍC1209, 2003.

[157] P. Simard, Y. LeCun, J. S. Denker, and V. B. Transformation invariance in pattern recognition-tangent distance and tangent propagation. In *Neural Networks: Tricks of the Trade*, 1996.

[158] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011.

[159] Q. Sun, Y. Lu, and S. Sun. A visual attention based approach to text extraction. In *ICPR*, pages 3991–3995, 2010.

[160] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[161] R. Tao, E. Gavves, C. G. M. Snoek, and A. W. M. Smeulders. Locality in generic instance search from one example. In *CVPR*, 2014.

[162] G. Tolias, Y. Avrithis, and H. Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *ICCV*, 2013.

[163] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):169–191, 2003.

[164] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010.

[165] S. S. Tsai, D. Chen, H. Chen, C.-H. Hsu, K.-H. Kim, J. P. Singh, and B. Girod. Combining image and text features: a hybrid approach to mobile book spine recognition. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1029–1032.

ACM, 2011.

[166] S. Uchida, Y. Shigeyoshi, Y. Kunishige, and Y. Feng. A keypoint-based approach toward scenery character detection. In *ICDAR*, pages 819–823, 2011.

[167] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.

[168] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE TPAMI*, 2010.

[169] J. Van de Weijer, T. Gevers, and A. D. Bagdanov. Boosting color saliency in image feature detection. *TPAMI*, 28(1):150–156, 2006.

[170] J. van de Weijer, T. Gevers, and J. M. Geusebroek. Edge and corner detection by photometric quasi-invariants. *TPAMI*, 2005.

[171] J. C. van Gemert. Exploiting photographic style for category-level image classification by generalizing the spatial pyramid. In *ICMR*, 2011.

[172] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *ICCV*, 2007.

[173] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *ICCV*, 2009.

[174] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *TPAMI*, 2012.

[175] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*, 2016.

[176] G. Wang, D. Hoiem, and D. Forsyth. Building text features for object image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1367–1374. IEEE, 2009.

[177] H. C. Wang and M. Pomplun. The attraction of visual attention to texts in real-world scenes. In *CogSci2011*, 2011.

[178] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *ICCV*, 2011.

[179] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng. End-to-end text recognition with convolutional neural networks. In *ICPR*, 2012.

[180] Y. Wang and D. Samaras. Estimation of multiple illuminants from a single image of arbitrary known geometry. In *ECCV*, pages 272–288. Springer, 2002.

[181] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. In *ECCV*, 2012.

[182] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD birds 200. 2010.

[183] L. Wu, P. Shivakumara, T. Lu, and C. Tan. A new technique for multi-oriented scene text lines detection and tracking in video. *Multimedia, IEEE Transactions on*, 2015.

[184] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, 2015.

[185] P. Xu, F. Davoine, and T. Denoeux. Evidential combination of pedestrian detectors. In *BMVC*, 2014.

[186] V. Yanulevskaya and J. M. Geusebroek. Significance of the weibull distribution and its

sub-models in natural image statistics. In *International Conference on Computer Vision Theory and Applications*, 2009.

[187] B. Yao, G. Bradski, and L. Fei-Fei. A codebook-free and annotation-free approach for fine-grained image categorization. In *CVPR*, 2012.

[188] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei. Action recognition by learning bases of action attributes and parts. In *ICCV*, 2011.

[189] C. Yao, X. Bai, and W. Liu. A unified framework for multioriented text detection and recognition. *Image Processing, IEEE Transactions on*, 23(11):4737–4749, 2014.

[190] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu. Detecting texts of arbitrary orientations in natural images. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1083–1090. IEEE, 2012.

[191] C. Yi and Y. Tian. Text string detection from natural scenes by structure-based partition and grouping. *Image Processing, IEEE Transactions on*, 20(9):2594–2605, 2011.

[192] C. Yi and Y. Tian. Text string detection from natural scenes by structure-based partition and grouping. *TIP*, 20(9):2594–2605, 2011.

[193] C. Yi and Y. Tian. Scene text recognition in mobile applications by character descriptor and structure configuration. *IEEE Transactions on Image Processing*, 23(7):2972–2982, 2014.

[194] L. Zhang, Z. Lei, Z. Guo, and D. Zhang. Monogenic-lbp: A new approach for rotation invariant texture classification. In *ICIP*, 2010.

[195] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, 2014.

[196] R. Zhang, P. sing Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survey. *TPAMI*, 21:690–706, 1999.

[197] Y. Zhang, X.-s. Wei, J. Wu, J. Cai, J. Lu, V.-A. Nguyen, and M. N. Do. Weakly supervised fine-grained image categorization. *arXiv preprint arXiv:1504.04943*, 2015.

[198] Q. Zheng and R. Chellappa. Estimation of illuminant direction, albedo, and shape from shading. In *CVPR*, 1991.

[199] Q. Zhu, M.-C. Yeh, and K.-T. Cheng. Multimodal fusion using learned text concepts for image categorization. In *Proceedings of the 14th annual ACM international conference on Multimedia*, pages 211–220. ACM, 2006.

[200] Y. Zhu, J. Zhu, and R. Zhang. Discovering spatial context prototypes for object detection. In *ICME*, 2013.

[201] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.

Samenvatting

Met dit proefschrift richten we onze aandacht op object herkenning en detectie voor een beter begrip in afbeeldingen. We onderzoeken afbeeldingen op het gebied van fundamentele elementen voor wat een persoon nodig heeft om te kunnen zien (dat wil zeggen; licht, objecten en een hoog niveau van semantische interpretatie).

Hoofdstuk 2: Positie van de lichtbron schatten
In hoofdstuk 2 richten we ons op het schatten van de positie van de lichtbron. Een veelgebruikte methode voor het schatten van de lichtbron positie (LP) gaat uit van de wet van Lambert. Echter, in de praktijk geldt deze wet niet voor alle verschillende soorten oppervlakten. In hoofdstuk 2 benutten we de invloed van oppervlakte kenmerken op de nauwkeurigheid bij het schatten van de LP. Gegeven een $RGB-D$ afbeelding, analyseren we eerst het effect van fotometrische (bijv. glanzend, mat) en geometrische (bijv. gebogen, vlak) oppervlakte eigenschappen. Hierna worden de oppervlakte eigenschappen gerangschikt met een supervised leertechniek. De volgorde wordt gebruikt voor het bepalen van de toegevoegde waarde van elk oppervlakte in de afbeelding met betrekking tot het schatten van de LP. Zodoende worden oppervlakte segmenten met de juiste fotometrische (d.w.z. Lambertiaanse reflectie) en geometrische oppervlakte eigenschappen hoger gewaardeerd. Bovendien gaan we er van uit dat de lichtbron, ten opzichten van de statische objecten in de scène, niet veranderd gedurende de opnamen van de video. Het enige dat veranderd is de relatieve positie van de lichtbron ten opzichte van de camera. Daarom stellen we voor om de camerapositie te gebruiken voor het beperken van het mogelijke posities van de LP in opeenvolgende frames in een RGB-D video frames. Om dit te bereiken, schatten we eerst de camera positie om zo de relatieve positie tussen de frames te verkrijgen. Daarna worden de camera posities gebruikt voor het optimaliseren van de LP voor de $RGB-D$ video frames. Experimenten op Boom en een door ons verzamelde dataset met $RGB-D$ video's, laten zien dat actuele methodes worden overtroffen door de voorgestelde methode. De resultaten laten zien dat het toekennen van verschillende gewichten aan oppervlakte segmenten op basis van hun eigenschappen beter werkt dan de actuele methodes waarbij alle oppervlakte segmenten evenveel bijdragen bij het schatten van de LP. Om precies te zijn verminderd de voorgestelde methode de hoekfout bij het schatten van de lichtbron van 12.9° naar 8.6° voor Boom, en van 12.9° naar 7.1° voor de $RGB-D$ video dataset. Bovendien, door gebruik te maken van de

camera positie voor het beperken van mogelijke posities van de LP in opeenvolgende frames, kan de nauwkeurigheid nog verder verbeterd worden naar 6.0° in vergelijking met 7.1° waarbij afzonderlijk frames gebruikt zijn.

Hoofdstuk 3 :Metrisch stelsel leren per beeld-stukje voor het bepalen van beeldafstanden
In hoofdstuk 3 richten we ons op het verbeteren van methodes voor het berekenen van de afstand tussen afbeeldingen met behulp van lokale beeld representaties. De bestaande methodes op dit gebied houden doorgaans rekening met alle of juist geen van alle variaties die mogelijk zijn. Echter, hierdoor kunnen de algoritmes minder goed fijne details onderscheiden. In hoofdstuk 2 stellen we een metrisch stelsel leer-methode voor die robuust is voor een gedeelte van deze variaties. Met deze methode kan vooraf bepaald worden voor welke gedeeltes de methode robuust moet zijn, waardoor we de juiste balans tussen de invariantie en onderscheidend vermogen kunnen afstellen.

We stellen een methode voor die robuust is voor beeld-stukjes. We laten het systeem een afstandsstelsel leren voor het meten van de afstanden tussen de beeld-stukjes. Dit gebeurd onafhankelijk voor alle beeld-stukjes door telkens een willekeurige transformatie te kiezen uit een verzameling van relevante beeld-transformaties. Deze transformaties geven voorkennis met betrekking tot het gedrag van de beeld-zoekopdracht na het toepassen van de transformatie in de kenmerkruimte. Voor het leren van het afstandsstelsel stellen we twee methodes voor: (i) volledig en (ii) direct. De volledige methode genereert synthetische beeld-stukjes, vervolgens berekend het de representatie per beeld-stukje, en wordt zodoende robuust door een afstandsstelsel te leren op deze beeld representaties. In tegenstelling tot de nogal inefficiënte aanpak van de volledige methode, in de directe methode stellen we een benadering voor waarbij de covariantie geschat kan worden van één enkel beeld-stukje. We genereren een transformatie-indeling slechts één keer, en gebruiken deze indeling om direct het afstandsstelsel voor een beeld-stukje te schatten, zonder dat er hiervoor synthetische afbeeldingen gegenereerd hoeven te worden. Het zoeken naar vergelijkbare beeld-stukjes voor een beeld-zoekopdracht wordt uitgevoerd met het nieuwe afstandsstelsel. De resultaten op de ALOI dataset laten zien dat de voorgestelde methode SIFT verbetert met 6.22% op het gebied van geometrische transformaties, en 4.43% voor fotometrische transformaties.

Hoofdstuk 4: Tekst detectie voor fijnkorrelige classificatie en logo retrieval
In hoofdstuk 4 richten we ons op fijnkorrelige object classificatie, gebruikmakend van teksten die in de afbeeldingen voorkomen. In tegenstelling tot actuele methodes die zich richten op alleen visuele aspecten, stellen wij een methode voor dat zowel de visuele als tekstuele aspecten gebruikt. Door gebruik te maken van de tekstuele aspecten, voegen we semantische informatie toe tijdens het herkenningsproces wat dieper gaat dan enkel visuele aspecten. Het succes van de voorgestelde methode hangt sterk af van de nauwkeurigheid waarin de tekstuele informatie dat uit de afbeeldingen vergaard kan worden. Om deze reden stellen we een robuuste letterteken lokalisatie methode voor, en en methode om tekstuele informatie uit afbeeldingen te verkrijgen. In tegenstelling tot actuele tekst detectie methodes, richten wij ons meer op de achtergrond dan op plekken waar tekst zich bevindt (voorgrond). Wanneer er een gebied met tekst is gedetecteerd, wordt deze nog door twee andere methodes behandeld voor tekst herkenning zoals ABBYY, een commerciële OCR oplossing, en een actuele letterteken herkenningsalgoritme. Hierna worden

de lettertekens omgezet in woorden, gebruikmakend van bi- en trigrams om de meeste waarschijnlijke volgorde van lettertekens te kiezen. In de laatste stap worden de textuele en de visuele aspecten gecombineerd voor fijnkorrelige classificatie. De voorgestelde methode is gevalideerd op vier publieke datasets: ICDAR03, ICDAR13, Con-Text en Flicker-logo. We verbeteren de actuele eind tot eind letterteken herkenningsalgoritmes met een marge van 15% op ICDAR03. Met dit hoofdstuk laten we zien dat het gebruik van tekstuele informatie uit afbeeldingen nuttig is in combinatie met visuele informatie voor fijnkorrelige classificatie, zo verbeterd de combinatie van tekst en visuele aspecten de classificatie van 60.3% naar 70.7%. Hetzelfde geldt voor het logo retrieval, waarbij we aantonen dat de combinatie een verbetering teweeg brengt van 54.8% naar 57.4%.

Hoofdstuk 5: Woorden in afbeeldingen zijn belangrijk voor de classificatie en retrieval
In hoofdstuk 5 maken we gebruik van woorden voor fijnkorrelige classificatie en logo retrieval, dat in tegenstelling tot hoofdstuk 4, waarbij alleen lettertekens zijn gebruikt. Bij het detecteren van woorden in afbeeldingen stellen we een generieke en volledig unsuperviced leertechniek voor, dat de locatie van de woorden in een afbeelding kan detecteren. In tegenstelling tot de actuele methodes voor tekst detectie die zich voornamelijk richten op het behalen van een hoge f-score, is de voorgestelde methode ontwikkeld met het oogmerk op een hoge recall. Een hoge recall is vereist voor de volgende stap in het algoritme, omdat lettertekens die niet herkend worden, ook niet gebruikt kunnen worden bij het vormen van woorden. Helaas is er geen methode die het beste is voor alle toepassingen van tekst detectie vanwege de grote variatie in lettergrootte, stijl en oriëntatie. Daarom stellen we voor om verschillende actuele letterteken detectie methodes te combineren. Om robuust te zijn tegen de verschillende omstandigheden waarin de afbeeldingen verkeren, maken we gebruik van kleur ruimtes met fotometrische invariante eigenschappen zoals robuustheid tegen schaduwen, lichteffecten en reflecties. De gedetecteerd tekst locaties worden gebruikt als invoer voor woord-herkenningsalgoritmes om de lettertekens om te zetten in woorden.

De resultaten laten zien dat het gebruik van tekstuele informatie de MAP van fijnkorrelige classificatie verhoogt van 60.3% naar 74.5%, en voor logo retrieval van 58.4% naar 62.7%. Bovendien leidt het gebruik van woorden tot een verbetering van 33.1% in fijnkorrelige classificatie, dat ten opzichten van 28.4% wanneer alleen lettertekens gebruikt zijn zoals in hoofdstuk 4.

Hoofdstuk 6: Het combineren van object detectoren
In hoofdstuk 6 richten wij ons op het detecteren van objecten in afbeeldingen door verscheidene actuele object detectie algoritmes te combineren. Er zijn in het verleden veel verschillende object detectie algoritmes voorgesteld, echter, gaan zij uit van bepaalde aannames in de manier waarop objecten in de afbeelding voorkomen en beeldomstandigheden. Het gevolg hiervan is dat geen van deze algoritmes breed toepasbaar is. In hoofdstuk 6 richten we ons op het vraag hoe verschillende object detectoren geselecteerd en gecombineerd kunnen worden. We maken gebruik van bekende object detectoren zoals DPM, CN en EES voor het detecteren van object locaties en de bijbehorende scores.

Experimenten op de PASCAL VOC07 en VOC10 datasets laten zien dat het combineren van verscheidene object detectoren duidelijk hoger scoort dan wanneer er maar één object detector wordt gebruikt. De verbeteringen op de VOC07 dataset zijn als volgt: DPM (8.4%), CN (6.8%) en EES (17.0%), en voor VOC10: DPM (6.5%), CN (5.5%) en EES (16.2%). Met deze experimenten laten we zien dat er geen belemmeringen zijn voor het type detector. Met de voorgestelde methode behalen we een verbetering van 2.4% ten opzichten van actuele object detectoren zoals RCNN op VOC07 wanneer RCNN gecombineerd wordt met andere detectoren die gebruikt zijn in hoofdstuk 6.

Acknowledgement

It was 6 years ago. I came from Norway to Amsterdam to do my master thesis. It was the day of the heaviest snowfall in Amsterdam in the past decade. All public transportation was canceled. I was walking with my heavy luggage, with my life packed in it. I did not know where to go or what to do. Starting a new life in a city with nobody that I know. No one to call for help. I remember the moment I asked myself: What am I doing here? Looking back now, I asked myself this question a couple of more times during the sleepless nights of my PhD. I am glad I have the answer today. As Frederick Douglass said "if there is no struggle, there is no progress". I made it! Maybe the road was bumpy and muddy but it was totally worth it. I feel fortunate to have meet many wonderful people who helped me throughout the journey over the past years. Now it is time to remember these dear friends.

Dear Theo, my journey with you started even before we met. Yes, I was reading your papers during my master studies. At the time I did understand that you made top-notch research, yet I did not know how truly *COLORFUL* personality was standing behind those titles. I really feel fortunate to have you onboard during my PhD journey. You have assisted me not in research only but also in the full spectrum of life. No matter how busy schedule you have, I always felt as I was on top of your priority list. Actually, I did not even need to be on the list, I would just knock on your door or call you (literally 7/24) to discuss anything. Our discussions were never one dimensional, with you always finding a different way to look at things. Although being creative and innovative is an important aspect of research, you taught me how to make research in *DEPTH*. In the football field, you keep insisting to position myself with respect to only one player of the opponent team so that you would back me up with the others. Indeed, you always got my back, on football court and outside of it. This gave me confidence to step up, do more and jump out of my comfort zone. With your help finding the light source in a room was not that difficult but more importantly you have taught me to be patient and see the light at the end of the tunnel. For all your guidance, patience and help, I cannot thank you enough. I am looking forward to the next phase of our 3DU adventure! And of course, a big thanks to your family for welcoming me all the time.

I would also like to thank Prof. dr. ir. Arnold Smeulders for being my co-promotor. I had pleasure to witness his wisdom while collaborating on several papers. His knowledge and vision are always inspiring in one way or another.

My sincere gratitude goes to Prof dr. Alain Trémeau for giving me the opportunity to study in CIMET and to support me throughout master studies which lead me to start my PhD; to Dr. Cees Snoek for providing me valuable advices whenever needed; to dr. Stevan Rudinac for positive spirit and many enjoyable conversations. I would also like to thank to Prof. dr. Maarten de Rijke and Prof. dr. Alan Hanjalić for participating in my dissertation committee.

I would like to thank Jan van Gemert for his guidance and incredible patience. You played an indispensable role in my PhD development. Dear Hamdi Dibeklioğlu, you were there for me from day one. You have patience and answers to all my questions. You always help me to work off my concerns and worries.

A big thanks goes to the people that I do not simply share the working environment but also share lots of important moments in life, Fares Alnajar, Ran Tao and Svetlana Kordumova. Memories of days and nights will always be reminding me of our endless conversations and laughters.

Additionally, my sincere gratitude goes to the people with whom I closely worked to build the chapters of this thesis or other research papers, Theo Gevers, Jan van Gemert, Ivo Everts, Yang Liu, Ran Tao and Silvia-Laura Pintea.

Yang, thank you for your loyalty and friendship. No matter what, when and where, you are always there for me with full support. Morris, you have made signature in almost all chapters of this thesis and cover with your impressive graphical skills. I am grateful for all your help. Dennis thanks for being such a positive and generous person. Efstratios, I always enjoy exchanging ideas with you. Specific thanks for the RCNN boxes which helped chapter 6. The list goes on and on: Dung, Emrah Taşlı, Albert Ali Salah, Masoud, Amir, Zhenyang, Roberto, Thomas, Jörn, Pascal, Spencer (thanks for English editing), Kandan, Jan Zahalka, Mihir, Sightcorp gangs (Aziz, Jasper, Diego and Sofia) and many others with whom I always have enjoyable conversations. PhD would not be the same without you guys!

My deep gratitude goes to the people behind the scenes who were always there to answer all my silly questions: Virginie Mes, Erik Hitipeuw, Betty Bijl and Guus Delen. Your help made my life a lot easier in the past years.

I would like to thank to "La ISLA Bonita" beach volley team members, Masoud, Amir, Ivo and Theo. It was a pleasure to win victories with you :) And also big thanks to "Dudes Arbide" and "Blauw-Wit" futsal team members, Gerben, Mark, Richard, Jan, Ivo, Apo, Dick, Nando and Theo. These social distractions meant a lot to me (especially discussions after game at Oerknal).

Dear 3DUniversum team (Theo, Yang, Morris, Finde, Daniel, Minh, Elena and Ray), thanks for coping up with me during challenging times of company and PhD life. It was not that easy to start a new challenge while there was another big one (PhD). Your help made this process a lot easier.

Dear Alper, Dervis, Gamze and Oktay, you make Amsterdam feel like home. Leaving birth country brings longing for own language, jokes, support and brotherhood. You are one of the best gifts Amsterdam gave me. You guys still make me feel truly myself so far from home. We have collected a lot of beautiful memories that make me smile every time I think of them. Amsterdam would not have been the same without you!

Dear Branko, Jagoda, Tanja, Srdjan, Arnold and Loes, thank you all for a warm welcome to the Netherlands. Your presence always makes us feel comfortable, accepted and home.

Draga porodice Hadziosmanovic, hvala to ste me toplo prihvatili od samog pocetka. Sad se vec cini da sam nekako oduvijek dio vas. Danas bih vam se zelio zahvaliti na povjerenju i strpljenju koje imate za Dinu i mene. U svim ovim godinama razdvojenosti od porodice, Vasa podrska je bila kriticna osnova za nas da i dalje vjerujemo u nase izbore, i u nas.

My love, Dina, this is the third diploma since you entered my life. I have to admit this one was the toughest. I am sorry for all the things that I have delayed. I am so thankful you stood by me all these years and loved me the way I am. Nobody deserves more than you to see the success of this journey. I could not have dreamt such a loving and caring partner like you. Your presence is my biggest medicine, my better half who reminds me of actual values in life. This thesis would not be finished, or may not have even been started without you.

Canim ailem, en kiymetlilerim, bu tez tamamen sizin eseriniz. Sizden aldigim guc ve ilham sayesinde bugun tezimin son satirlarini yaziyorum. Sevginiz icimde, gittigim her yere yanimda goturdum. Sizden ayri gecirdigim dakikalari, saniyeleri hic bir sey telafisi edemez ama elimden geldigince size duydugum minneti anlatmaya calisayim.

Canim babam, aldigim kararlarin kucugunden buyugune her zaman saygi duyup, yanimda durdugun icin sana ne kadar tesekkur etsem azdir. Kucuklugumden bu yana vermis oldugun nasihatler ve egitim sayesinde karsima cikan zorluklari asmam cok da zor olmadi. Ayrilik, ozlem elbette zor ama, armut dibine dusermis. Sen kendi hayatini baska bir sehirde, her seye sifirdan baslayarak, gece-gunduz demeden calisarak kurdun. Benden de farkli bir sey yapmam beklenemezdi. Yaptiklarima soyle bir baktigin zaman senin hayatinin farkli yillarda yasanmis bir versiyonu olarak gorebilirsin. Aradaki tek fark her adimimi dustugum zaman elimden tutacak bir babam oldugunu bilerek attim.

Canim annem, vazgecilmezim, koruyucu melegim. Bazen sabahlara kadar basimda bekledin bazen de sabahlara kadar yolumu gozledin, yanimda olamadigin zamanlarda ise dualarinla beni destekledin. Elde ettigim basarilarla her ne kadar mutlu olsanda, senin icin en onemli olan seyin ben oldugumu her zaman hissettirdin. Bana verdigin sinirsiz sevgi ve emegin karsiligini vermek imkansiz, iyi ki varsin!

Canim ablam, diger yarim, anne ve babamin paha bicilmez mirasi, aramizda ne kadar mesafeler olsada kalplerimiz hep bir atti. Varligin her zaman bana guc oldu. Hayatima kattigin butun guzellikler icin ne kadar tesekkur etsem azdir. Canimdan otesin, iyiki varsin. Engin, aci ve tatli gunumuzde yanimizda oldugunu her zaman hisettirdin. Her sey icin cok tesekkur ederim. Ayrica evinize her geldigimde o kadar sicak karsiladin ki artik Tarsus'a gitmez oldum: Asya'm, ilk goz agrim, biricigim, hayatindaki bir cok ilkte yaninda olamadim ama bundan sonra cok daha fazla birlikte gecircek zamanimiz olcak. Doktoranin zor gunlerinde hayatima renk ve heyecan getirdin.

Sezer Karaoğlu

October 2016

The best way to
make your dreams come true
is to wake up