Eindhoven University of Technology

MASTER

Prediction and Improvement of the Outcomes of Image Recognition Algorithms Applied to an Automated Invoice Processor

Artsyman, I.

Award date:
2020

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Technische Universiteit **Eindhoven** University of Technology

Department of Mathematics and Computer Science

Prediction and Improvement of the Outcomes of Image Recognition Algorithms — Applied to an Automated Invoice Processor

Master Thesis

Ilya Artsyman

Supervisors:
Thesis supervisor: P. J. De Andrade Serra
Company supervisor: D. Mocking
Committee members: R. Pires da Silva Castro, V. Menkovski

Eindhoven, August 2020

Contents

Chapter 1

Introduction

Insurance companies nowadays are aiming for digitalization of reimbursement process for medical treatments. This makes reimbursement faster. For this purpose an insurance company has decided to contact ORTEC in order to work on a digital solution. ORTEC is a consultancy company which specializes in applying mathematical knowledge to industrial and business problems.

Before collaboration with ORTEC, in this insurance company the reimbursement process required a lot of manual work. Clients sent photos or scans of invoices for their treatments and they were checked by employees in order to retrieve information. ORTEC worked on automatization solutions for the dental care sector and came up with an algorithm for this problem. This algorithm analyzes an image and extracts relevant information from it. However, this algorithm not always provides with complete information. Moreover, it takes roughly 10 seconds to analyze an image and return either extracted information or a message saying that the required elements of an invoice were not recognized. The latter situation contains cases like “there was no total amount detected” and “there was no text detected” which are often caused by low image quality. In these cases, a user is requested to take another photo of the invoice. Thus, in the cases when the required elements of an invoice were not recognized due to insufficient image quality, the long run time of 10 seconds, which is actually waiting time for the user, is not affordable.

The graduation project is aimed at improving the existing procedure, suggested by ORTEC, by development of new algorithms for image processing. The first goal is to develop an algorithm which will promptly tell clients if their image is good for further recognition or if it should be taken again. Second goal is to come up with image pre-processing techniques which will improve recognition percentage of images and will be chosen depending on an image.

In order to approach the first goal, the research on image characteristics evaluation and image quality assessment is performed to select and implement features extraction methods relevant to this project. Afterwards, machine learning algorithm is chosen and applied to these features in order to predict the existing algorithm outcome. For the second goal, current research on image enhancement methods is studied and the relevant methods are combined with feature extraction part in order to construct an algorithm for the choice of relevant image enhancement techniques for a specific image.

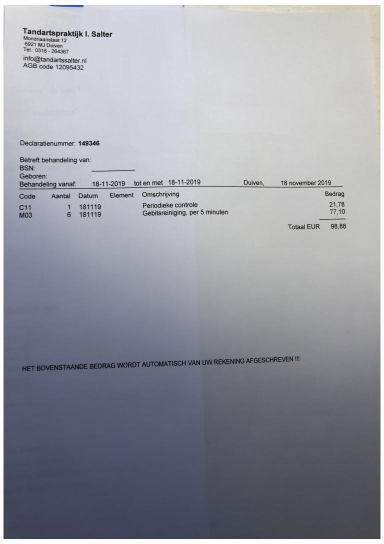
The report is structured as following: in the Chapter 2 structure of the data is described as well as its availability. In the Chapter 3 the existing algorithm developed by ORTEC is described and the project problem is discussed in more details. It is followed by the Chapter 4 with literature review. In the Chapter 6 possible approaches to the problem are discussed. Then, in the Chapter 5 a process of data labelling and preparation for the project is described. Afterwards, in the Chapters 7 and 8 the first goal of the project is approached split into features extraction part and classification part. Finally, in the Chapter 9 the second goal of the project is approached.

Chapter 2

Data description

The images dataset consists of images of invoices from dental care. The images can represent photos of paper invoices, photos of laptop or phone screens with invoices, or scans of invoices. The photos vary in orientation, illumination, paper condition, and image quality and may include handwritten text alongside with printed. The images are usually rectangular and their typical size is 2500×3200.

In this project, only anonymized images can be used for the analysis due to privacy. There are 529 distinct anonymized images available. Actual invoices that are analyzed by the ORTEC algorithm are not anonymized as client details are important for reimbursement. Thus, the performance of the developed algorithm on practice might differ from the performance on the anonymized dataset.



Tandartspraktijk I. Salter
Mondriaanstaat 12
6921 MJ Duiven
Tel.: 0316 - 264367
info@tandartssalter.nl
AGB code 12095432

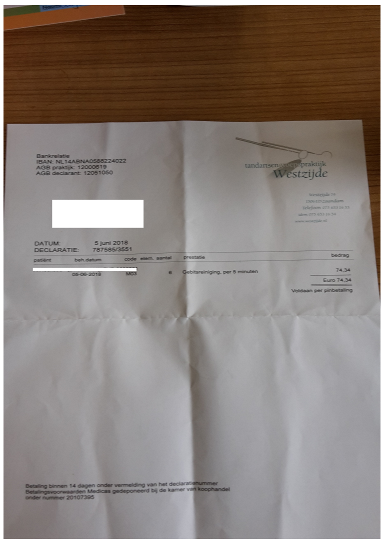
Declaratienummer: **149346**

Betreft behandeling van:
BSN:
Geboren:
Behandeling vanaf: 18-11-2019 tot en met 18-11-2019 Duiven, 18 november 2019

Code	Aantal	Datum	Element	Omschrijving	Bedrag
C11	1	181119		Periodieke controle	21,78
M03	6	181119		Gebitsreiniging, per 5 minuten	77,10
				Totaal EUR	98,88

HET BOVENSTAANDE BEDRAG WORDT AUTOMATISCH VAN UW REKENING AFGESCHREVEN !!!

(a) Anonymization with pieces of the invoice.



Bankrelatie
IBAN: NL14ABNA0588224022
AGB praktijk: 12000619
AGB declarant: 12051050

tandartsengroepspraktijk
Westzijde

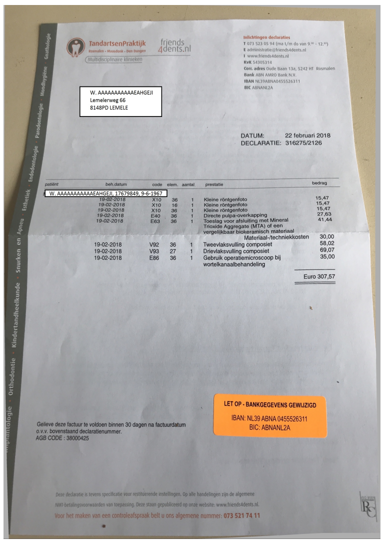
DATUM: 5 juni 2018
DECLARATIE: 787585/3551

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
	05-06-2018	M03		6	Gebitsreiniging, per 5 minuten	74,34
						Euro 74,34

Voldaan per pinbetaling

Betaling binnen 14 dagen onder vermelding van het declaratienummer
Betalingsvoorwaarden Medicas gedeponeerd bij de kamer van koophandel
onder nummer 20107395

(b) Anonymization with white boxes.



TandartsenPraktijk
Rosmalen - Maasdonk - Den Dungen
Multidisciplinaire klinieken
friends4dents.nl

Inlichtingen declaraties
T 073 523 05 94 (ma t/m do van 9.00 - 12.00)
E administratie@friends4dents.nl
I www.friends4dents.nl
KvK 54305314
Corr. adres Oude Baan 13a, 5242 HT Rosmalen
Bank ABN AMRO Bank N.V.
IBAN NL39ABNA0455526311
BIC ABNANL2A

W. AAAAAAAAAAAEAHGEJI
Lemelerweg 66
8148PD LEMELE

DATUM: 22 februari 2018
DECLARATIE: 316275/2126

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
W. AAAAAAAAAAAEAHGEJI, 17679849, 9-6-1967						
	19-02-2018	X10	36	1	Kleine röntgenfoto	15,47
	19-02-2018	X10	16	1	Kleine röntgenfoto	15,47
	19-02-2018	X10	36	1	Kleine röntgenfoto	15,47
	19-02-2018	E40	36	1	Directe pulpa-overkapping	27,63
	19-02-2018	E63	36	1	Toeslag voor afsluiting met Mineral Trioxide Aggregate (MTA) of een vergelijkbaar biokeramisch materiaal	41,44
					Materiaal-/techniekkosten	30,00
	19-02-2018	V92	36	1	Tweevlaksvulling composiet	58,02
	19-02-2018	V93	27	1	Drievlaksvulling composiet	69,07
	19-02-2018	E86	36	1	Gebruik operatiemicroscoop bij wortelkanaalbehandeling	35,00
						Euro 307,57

Gelieve deze factuur te voldoen binnen 30 dagen na factuurdatum
o.v.v. bovenstaand declaratienummer.
AGB CODE : 38000425

LET OP - BANKGEGEVENS GEWIJZIGD
IBAN: NL39 ABNA 0455526311
BIC: ABNANL2A

Implantologie · Orthodontie · Kindertandheelkunde · Snurken en Apneu · Esthetiek · Endodontologie · Parodontologie · Mondhygiëne · Gnathologie

Deze declaratie is tevens specificatie voor restituerende instellingen. Op alle handelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing. Deze staan gepubliceerd op onze website: www.friends4dents.nl.
Voor het maken van een controleafspraak belt u ons algemene nummer: 073 521 74 11

(c) Anonymization with white boxes with fake personal details.

Figure 2.1: Examples of anonymized invoices images.

Anonymization is done by covering personal information of clients in invoices (such as name, BSN and birth date) with blank pieces of an invoice copied from an image of this invoice

(Fig. 2.1a), with empty white boxes (Fig. 2.1b), or with white boxes which contain fake personal details (Fig. 2.1c). This introduces some distortions into images and, hence, might cause a difference in the performance on the anonymized dataset and on the actual images which are analyzed by the ORTEC algorithm.

The invoices consist of so-called fields. They include client details, dentist details, number (ID) of the invoice, total amount to be paid, and fields related to treatments. Treatments are usually represented as a table with a treatment description, code, amount, date, and price. Some treatments also include an indication of the jaw (top or bottom) and the element number (tooth number) for which the treatment was made as this information can influence the price of treatments. The structure of the table can be clearly seen in Fig. 2.1c, where the columns are present for a treatment date ("beh. datum"), a treatment code ("code"), an element number ("elem."), amount of treatments ("aantal"), a treatment name ("prestatie") and a price ("bedrag"). Indication of the jaw is not necessary for any of the treatments in this example and, hence, it is not present in the invoice. An invoice may also include other information (e.g., contact details of a medical center) which is not relevant for reimbursement. Thus, the invoices usually have similar fields but their location and structure of the invoice might be different. All of the invoices are in Dutch language but for convenience, we refer to fields and information in an invoice in English.

Chapter 3

Current algorithm discussion

In this chapter, we will discuss the existing ORTEC algorithm and how it works.

ORTEC has developed an application, which a client uses to take a photo of an invoice and send it to the insurance company. It is implemented using Python programming language. In this application, the photo is resized to the pre-specified size and after that, the image is analyzed with the help of the Microsoft's OCR (Optical Character Recognition) API ([2]). It returns a .json file with coordinates of each text box on the image and recognized text in this box. For example, for a piece of an invoice as in Figure 3.1, part of an output of OCR looks like the following:

- Word: "Betreft:" [101,1533,131,33];
- Word: "behandeling" [247,1533,221,40];
- Word: "van:" [482,1541,75,24],

where the first two numbers are x- and y-coordinates of the top-left corner of the box and the last two are the width and the height of the box.

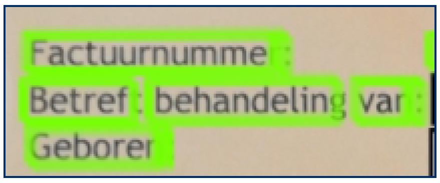


Figure 3.1: An example of text boxes recognized by OCR

The information extracted by OCR is used by the algorithm in order to extract necessary fields from the image. The words are combined into horizontal lines and vertical columns based on coordinates and sizes of text boxes. The pre-specified thresholds are used for uniting the boxes into the blocks. For example, the boxes are united into a horizontal line if the difference between their top edges or bottom edges is less than the threshold. The same methodology is used for vertical blocks with a difference that the distance is measured between the left and right edges of the boxes. The thresholds equal half of the text height for horizontal blocks and the height of the text for vertical blocks. After constructing the blocks, all the fields are located based on keywords (such as “Invoice number”, “Price” etc.). Then, the extracted information is searched for in a database to perform checks, for example: if the person exists or if the treatment has the specified price. If some fields are lacking from OCR, the database is also used: for example, the treatment code can be used to find the full treatment name and the price or other way round. Moreover, the total amount is also checked to be the sum of all the stated amounts.

There are mandatory fields that have to be recognized (such as client name, date of treatment, code of treatment and its price) and optional fields (for example, invoice number, and full name of treatment). There are also two fields which are mandatory only for certain treatments: jaw indication and element number. Based on this analysis, an image is assigned to a category “fully recognized” if all of the mandatory fields are recognized, “partially recognized” if 1-3 of the mandatory fields are not recognized, and “rejected” if more than 3 of the mandatory fields are not recognized.

Moreover, there are other reasons for the image to be rejected. The complete list of them is the following:

- There is no text on the image;
- The image is not recognized as an invoice;
- Too many fields (more than 3) are not recognized in the invoice;
- The total amount does not equal to the sum of all costs;
- The invoice is cropped, so not all information is visible;
- The invoice is already paid by the insurance company;
- The client is not insured in this insurance company;
- Several invoice IDs are found (which is not allowed by application);
- Only one page of a multiple-page invoice is found.

In case the image is not fully recognized, a tesseract algorithm is applied to boxes with unrecognized text in order to see if it can provide better extraction quality.

In the case of a partially recognized invoice, the client is asked to fill in the missing fields manually.

In case of rejection, the application also returns the reason for rejection from the list above. It can be seen that rejections might happen correctly and incorrectly. For example, when an image is not recognized as an invoice due to bad quality of the image, it is considered as an incorrect rejection and the respective image is categorized as “not recognized”. On the other hand, rejection due to being already paid by the insurance company is a correct rejection as it is supposed to be rejected. The same distinction is present between correctly and incorrectly partially recognized images. For example, an image is correctly partially recognized, when the date of one of the treatments is absent on the invoice. It is important to distinguish these cases when labelling the data.

In the ORTEC algorithm, Microsoft's OCR engine is preferred to Google's OCR due to some privacy issues even though Google OCR is considered to be more accurate by some sources as discussed in [1]. However, the algorithm which is used in Microsoft OCR is not public, hence, it is unknown, which parameters of an image have the largest influence on its performance.

The tesseract algorithm, on the other hand, is public. In the beginning, it used traditional computer vision algorithms for text recognition as discussed in [96] and later LSTM (long short-term memory) network was added in order to increase the accuracy of words recognition. It is a neural network that reads character by character in discovered text boxes and uses as input both new character and information obtained on the previous steps of the neural network. This way, it can learn, for example, language rules in order to distinguish the text.

The average running time of the ORTEC algorithm is around 11-12 seconds. This is unacceptable in the case the output of the algorithm is that invoice is not recognized. That is why in the project we are working on a classification algorithm that can be run before the ORTEC algorithm and will promptly tell a user that his photo will not be recognized. We aim at the algorithm which will work in 1 second.

Chapter 4

Literature review

In this project, we are developing two algorithms: one for prediction of the outcome of the ORTEC image recognition algorithm and another for image enhancement. In this chapter, we will provide a literature review of the relevant research for both algorithms.

For the prediction algorithm, we will use various image features. They can be split into two groups: features that can be extracted from any type of images and features which are specific for images of documents. These two groups are discussed in Sections 4.1 and 4.3 respectively.

Furthermore, OCR methods can be investigated in order to understand which characteristics of an image might influence OCR performance and which pre-processing techniques can be used. They are discussed in Section 4.2.

Finally, we review the image enhancement techniques in Section 4.4.

4.1 Image features extraction methods

Image recognition is closely related to image quality: the better the quality the higher is the chance the image will be correctly recognized. Besides this, some other image statistics which do not represent quality can be also used as features. Thus, in this section, we start with a discussion of image statistics that are not related to quality and then review the features related to image quality assessment (IQA).

4.1.1 Image statistics

Various image statistics that do not represent any qualitative feature of an image are proposed in the survey [86]. They are split into the following categories: color-related statistics, first-order, second-order, and higher-order statistics of an image. First-order statistics take into account properties of single pixels, second-order statistics incorporate dependencies between neighbouring pixels, and higher-order statistics account for larger dependency structures.

Histograms and statistical moments of the original or log-transformed images are suggested as first-order statistics. Gradients are proposed as second-order statistics. To obtain higher-order statistics, it is suggested to apply wavelet transforms, discrete Fourier transform (DFT) or principal component analysis (PCA). For instance, phase spectra variance can be obtained after whitening an image by means of PCA. Finally, several color schemes apart from RGB are suggested to capture color statistics.

4.1.2 Image quality assessment (IQA)

Image quality can be defined in different ways. For example, as a value of a single characteristic of an image, such as brightness or sharpness. On the other hand, an image quality metric might combine different image features to obtain a single quality score. Thus, in this section research in specific image features as well as in general IQA methods is reviewed. There exist several types of IQA methods depending on the availability of the original, or higher quality image. These methods are called full-reference (FR) if the original is available, partial reference if it is partially available and no-reference (NR) if it is not available. In our case, we are interested in NR IQA.

Besides this, IQA methods can be classified by some other properties (as discussed in [65]):

1. Necessity for a dataset to be subjectively evaluated by human:
   - **Opinion-aware** which require such a dataset;
   - **Opinion-unaware** which does not;
2. The distortions which are assumed:
   - **Distortion-specific** which assume only one distortion present in an image;
   - **General**;
3. Area of application:
   - **NSS** (natural scene statistics) methods which use image statistics and suit for assessment of different types of images;
   - **non-NSS** — trained on some specific images database and, hence, have the better performance on the images of the same type as in the training database.

There is a lot of research done recently on IQA that relies on deep learning (as in [66], [67]) and these methods are shown to perform better than non-deep learning ones but only on some datasets. Moreover, these methods require large training samples, hence, they are not feasible in our project. Thus, we will focus on IQA with the use of feature extraction.

Most of the IQA metrics require application of image transformations in order to extract features. Generally speaking, their construction can be fitted into the following scheme:

1. The image is converted into gray-scale (optional);
2. Some transformations are applied to the image (optional):
   (a) Spatial transformations (filters);
   (b) Spectral transformations (DFT, DCT, wavelets);
3. Features are extracted from the transformed image:

   (a) Mean, standard deviation or maximum of transformed image pixels is taken;
   (b) Some other, more complicated, function is applied to the transformed image;
   (c) Some distribution is fitted to transformed image pixel values and estimated parameters are taken as features;

4. Features are used to construct the quality (or another) metric:
   (a) If it is the only feature, it is used as metric;
   (b) Machine learning is applied to features based on subjective evaluated data;
   (c) Some function is applied to features (sum, average or something more complicated);
   (d) Some distribution is fitted to the features and it is compared to a fitted distribution to the same features of images which are considered to be of high quality.

The most used features are colors, brightness (luminance), contrast, sharpness (opposite to blur), and noise. Some other features which are extracted can't be interpreted as indicative of the quality of an image and are only defined mathematically as a function of an image. These functions are mostly complicated and not all of them are used in this project. Thus, some of them will be omitted and mentioned only as "a function".

The remaining part of this section will be split into subsections. Firstly, we discuss possible image transformations. Afterwards, IQA techniques related to specific image characteristics are discussed in separate subsections. The last subsection refers to general IQA techniques which are used to evaluate overall image quality without a connection to any specific image characteristic.

Image transformations

$$\begin{pmatrix} 0 & -1 & 1 \end{pmatrix}$$

Horizontal forward derivative filter

$$\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

Vertical Sobel filter

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

Laplacian filter

Figure 4.1: Examples of filters.

We can distinguish several groups of transformations which can be applied to an image as mentioned above. These are spatial and spectral transformations.

Among spatial transformations, the most popular ones are convolutional filters. These are matrices that are used to convolve an image. Examples are derivative filter, Sobel filter, and Laplacian filter which can be found in Fig. 4.1. When they are applied to an image, they are scaled by the sum of absolute values of matrix elements.

The spectral transformations of image $u(x, y)$ of the size $N \times M$ include DFT (eq. 4.1), discrete cosine transform (DCT) (eq. 4.2), and wavelet transforms. DFT of an image in practice is usually calculated using fast Fourier transform algorithm.

$$\hat{u}(k,l) = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} u(x,y)e^{-\left(\frac{2\pi ikx}{N}+\frac{2\pi ily}{M}\right)}. \tag{4.1}$$

$$\hat{U}(k,l) = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} u(x,y)\cos\left(\frac{\pi k}{N}\left(x+\frac{1}{2}\right)\right)\cos\left(\frac{\pi l}{M}\left(y+\frac{1}{2}\right)\right). \tag{4.2}$$

Wavelet transformations, as opposed to DFT, allow localizing frequencies in space. There exist various types of wavelet transformations: for example, Haar wavelet or Daubechies 9/7 or 5/3 wavelets ([26]). They all have the same idea. In wavelet transformation an image is passed through the series of low-pass filters $h_0[n]$ and high-pass filters $h_1[n]$. Firstly, the filters are applied to the original image: 4 different combinations of low-pass and high-pass filter to obtain 4 subbands. They are LL, HL, LH, and HH subbands, where “L” corresponds to a low-pass filter and “H” to a high-pass filter. The first mentioned filter is applied in the horizontal direction, second filter — in the vertical. Subsampling in the horizontal direction is applied between the filters and subsampling in the vertical direction afterwards. That results in vertical (HL subband), horizontal (LH subband) and diagonal (HH subband) edges discovery. These subbands are twice smaller than the original image due to double subsampling (Fig. 4.2). Afterwards, the whole process is repeated for the LL subband. The number of iterations is usually given in each paper.

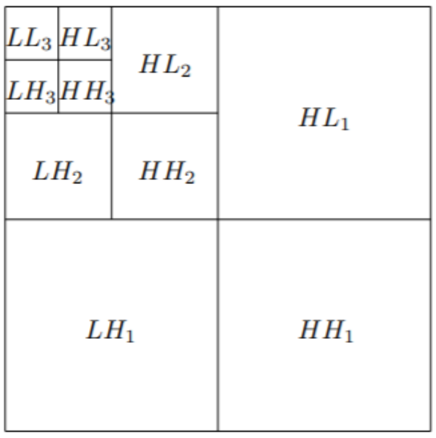


Figure 4.2: Subbands after application of wavelet transform.

Thus, the main question when constructing wavelets is which filters to choose. There exist several methods resulting in different types of wavelet transformations mentioned above. Theory behind wavelets and its applications to image processing can be found in the following papers and tutorials: [51], [22], [78] and [52].

Another filter which is often used in IQA and is closely related to wavelet transforms is a log-Gabor filter. It allows discovering edges not only in vertical, horizontal, or diagonal directions but also in other custom angles (central orientations). Log-Gabor filter in polar coordinates in frequency domain has the form as in eq. 4.3. This filtering method is discussed in [32].

$$G(\omega,\theta) = e^{-\frac{\log\left(\frac{\omega}{\omega_0}\right)}{2\sigma_r^2}} \cdot e^{-\frac{(\theta-\theta_j)^2}{2\sigma_\theta^2}}, \tag{4.3}$$

where σ_r and σ_θ are bandwidth constants and ω_0 and θ_j are parameters of the filter representing central frequency (scale) and orientation. Usually, set of log-Gabor transforms with N different ω_0 and J different $\theta_j = j\pi/J$ for $j \in \{0, \ldots, J-1\}$ is applied to an image. DFT of an image converted into polar coordinates is multiplied by log-Gabor filter element-wise in order to obtain filtered image.

One more transformation which is used by some authors of IQA methods is a 1D pseudo-Wigner transform (eq. 4.4). It is applied iteratively to windows of size N of a flattened image z. The image can be flattened in vertical or horizontal direction.

$$W(k,l) = \sum_{m=-N/2}^{N/2} z(k+m)z^*(k-m)e^{-2il\left(\frac{2\pi m}{N}\right)}, \tag{4.4}$$

where z^* is a complex conjugate of z.

Now, we can look into the methods of IQA where these transformations can be used.

Brightness

To begin with, let's look into brightness evaluation methods. In [6] several brightness metrics are proposed by applying several different functions to color values of pixels and taking the average, the maximum and the standard deviation of them as metrics of brightness.

Other sources of brightness evaluation are methods of transformation into grayscale. This is directly related to brightness as the gray value represents the brightness of the pixel. In [49] 13 different grayscale transformation methods are compared. All of them apply some function to each pixel separately (e.g., a linear combination of colors) to get a brightness map of the image. Thus, as an overall image brightness, the maximum or the average can be taken again.

Contrast

The next feature that we consider is contrast. For it RMS (root mean square) and Michelson metrics are among the ones that are often used as mentioned in [54]. Both of them use the grayscaled image and apply a function to it to get a contrast metric.

More IQA methods based on contrast evaluation are suggested in [110], [30] and [35]. In [30] an opinion-aware method is suggested, where for a training set of gray-scaled images the mean, the variance, and other simple features are calculated and then a normal distribution is fitted to them. The PDF values of these distributions on the evaluated images are used as features and support vector regression (SVR) is applied to get the final metric. SVR is a regression method which is based on the same principle as support vector machine but performs the regression instead of classification. In contrast to usual regression, it does not minimize the error but tries to fit the error into some boundaries.

In [110] an opinion-aware CEIQ (contrast-changed image quality) measure is proposed. There, as features for SVR entropy and cross-entropy of the gray-scaled image are used alongside with a similarity measure of the original and the contrast-enhanced images. In [35] NIQMC (NR image quality metric for contrast distortion) is proposed where a local and a global quality metrics are combined together to obtain an overall quality metric. To calculate the local measure, firstly, an image predicted by AR (autoregression model) is subtracted from the original. After that, the maximum of the local entropy values of the most salient regions (chosen as in [37]) of the resulted image is taken as measurement. As a global measurement, Jensen-Shannon divergence is used between the uniform and the image histograms.

Sharpness

There is a lot of research done on sharpness (or blur) metrics. In order to evaluate sharpness, spatial or spectral transforms are usually firstly applied to an image. In this section, we discuss various methods proposed in the literature.

To begin with, let's look into a simple method suggested in [81] where spatial transforms are used to evaluate sharpness. There, a Laplacian filter or a Sobel filter is applied to an image. Then, a magnitude function, the average, and the standard deviation of the transformed image are calculated as different sharpness metrics.

There also exist methods where spatial and spectral transforms are used together. This way, S3 method was introduced in [102], where the author uses the product of a spectral and a spatial feature to evaluate sharpness. Total variation, which is introduced in [9], is used as a spatial sharpness measure. Originally, it is defined as the sum of values over an image after the application of derivative filters. However, in [102] the image is split into equally-sized small square regions (e.g., 3×3 or 4×4 pixels) and the maximum of total variation over all these regions is taken as a spatial feature. This method of splitting the image into small regions of the same size and applying some transformation or a function to them (locally) is widely used in IQA. To get a spectral feature, the author also splits the image into regions. In this case, their size is 32x32 and they are overlapping. Then, DFT is applied to each region. Afterwards, a composition of a magnitude function, function minimization, and sigmoid function are applied to obtain a sharpness map and the maximum is taken as a spectral sharpness measure.

In [31] the authors suggest a new non-NSS opinion-aware metric JNB (Just Noticeable Blur) which is reported to perform better than all previously developed methods. There, a Sobel operator is applied to an image and the contrast is estimated in regions of the image with a sufficient number of edge points. To estimate it, the authors firstly performed an experiment where people evaluated the sharpness of images. Based on its results, they proposed a function to estimate contrast in each block. After that, the sharpness of the whole image is calculated from the blocks using a Minkowski metric. In [73] the authors propose an improved CPBD (Cumulative Probability of Blur Detection) metric, which uses JNB. There, the authors slightly change the contrast function and use 0.63 percentile of the local JNB metrics as a new overall sharpness metric. This metric is also implemented in the cpbd Python package. In [117] another quality metric based on JNB is proposed. It is improved to account for different noise levels in different regions of an image.

In opinion-unaware methods in [8] and [59] the idea of Global phase coherence, initially introduced in [9], is used to construct a sharpness metric with lower computational complexity. Firstly, the image has to be periodized and dequantized as described in [71] and [27] respectively. Afterwards, DFT is applied to the image. Finally, the sharpness index is calculated as a function of a gradient cross-correlation matrix obtained from DFT transformed image and total variation. In [59] another function is suggested to be applied to total variation and the gradient cross-correlation matrix to obtain a sharpness metric, so-called s-index.

In [41] an opinion-unaware method is proposed. There, DCT is applied to 8×8 regions of an image and for each coordinate pair (i, j) a Laplace distribution is fitted to DCT coefficients at these coordinates in all regions. Then, a function is applied to the parameters of the fitted distributions to get sharpness.

In [42] another opinion-unaware method is introduced. It starts with applying a wavelet transform with log-Gabor filter and then calculating the local phase coherence (LPC) map based on it. LPC is introduced in [106] and uses the fact that the phases of complex wavelet coefficients form a highly predictable pattern in the proximity of sharp edges. Then, as a sharpness metric, a weighted average of the LPC map across the image is taken.

Some papers suggest methods with a decreased computational complexity of calculations. For example, in an opinion-aware method from [105] features are constructed by applying combinations of linear and absolute value functions to the image filtered with a derivative filter. After that, a polynomial function is applied to these features in order to get sharpness measure. However, parameters of this polynomial function should be evaluated based on linear regression on a subjectively evaluated image dataset.

Another fast method is suggested in [103]. There, a Daubechies 9/7 wavelet transform with 3 levels of decomposition is applied to an image. After that, log-energy is computed for LH, HL, and HH subbands in each level and the total log-energy in each level is calculated as a linear function of the respective subbands. Finally, the sharpness index FISH is calculated as a weighted sum of the total log-energy values at each level.

One more opinion-unaware blur estimation metric with low complexity was suggested in [69]. There, a Sobel filter is applied to find edges and after that, the average width of edges is considered as a blur metric.

Sharpness and noise combined

Several characteristics of an image can be evaluated simultaneously in order to obtain an IQA metric. This way, sharpness and noise are sometimes combined.

For example, in [118] the authors calculate gradients by applying derivative filters and construct gradient matrices of regions of an image with rows referring to image pixels and columns — to a direction of the gradient. After that, they find maximum singular values for each of these local gradient matrices. Then, a square root of a quadratic function of the maximum singular value for each matrix is evaluated and its average over all regions is considered as a final sharpness and noise metric. In [119] an improvement of this metric is proposed. The previously mentioned metric required a reliable noise variance estimate which is not always easy to obtain. The improved metric suggests an estimate for this variance.

In an opinion-aware method in [56] a Daubechies 5 wavelet transform is applied locally to the blocks of size 64×64 and the variance of eigenvalues of the HH subband is calculated for each block. Then, their average is taken as a feature. The second feature is constructed by firstly applying the Wiener filter for denoising and, then, evaluating the LPC blur metric. These features are used to perform a linear regression of the quality based on the subjective scores.

In an opinion-unaware method in [21] two blur and two noise metrics are calculated for the image. Firstly, an image is filtered using an average filter. Then, the edges are detected by means of derivative filters both in the original image and in the filtered image. Afterwards, two blur metrics and two noise metrics are calculated as rational functions of the values of edge pixels in the original image and the filtered image respectively. Then, a linear combination of all 4 metrics is considered a final sharpness and noise metric.

Noise

Noise can be also used on its own in order to evaluate image quality. For instance, in [64] local gradient matrices are used only to evaluate the amount of noise. Their traces are calculated and assumed to follow a gamma distribution. After that, a two-step iterative procedure is performed until the convergence of the noise variance. Firstly, the patches with weak texture are chosen using some threshold and, then, the noise variance is estimated by means of PCA.

Another approach for evaluation of the amount of noise is proposed in [46]. There, the authors apply a combination of Laplacian filters to an image and estimate the noise variance as the sample variance function of the pixel values of the filtered image.

One more noise specific metric is the CINEMA (Content Independent Noise Estimation for Multimedia Applications) metric described in [98]. In this paper, regions of the image with weak texture are selected and a wavelet transform is applied to them. Finally, the minimum over the selected regions of the standard deviation of the difference between the original block and the LL subband is considered a noise metric.

General IQA

Now, let's look into general IQA methods. Firstly, we start with IQA methods mentioned in a survey [65]. Afterwards, we look into other available methods.

One of the most popular IQA methods is BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator) and it is introduced in [70]. BRISQUE is an opinion-aware NSS metric. In this method, Gaussian filters are applied to a gray-scale image in order to obtain locally normalized luminances. These luminances are used to produce image features by applying functions to them and then approximating parameters of a fitted generalized Gaussian distribution (GGD) and an asymmetric generalized Gaussian distribution (AGGD). Finally, SVR is applied to these features. This method is implemented in several packages in Python. However, its computational complexity is fairly high: it takes approximately 1,5 seconds for a 512×512 image according to authors (on a standard HP Z620 workstation with a 3.2GHZ Intel Xeon E5-1650 CPU and an 8G RAM).

Another popular metric IL-NIQE (Integrated Local Natural Image Quality Evaluator) is proposed in [114]. It is an opinion unaware metric and, in addition to the features described in BRISQUE, it uses gradient, orientation, and color statistics obtained after applying various transforms to an image. There are Gaussian filtering (blurring), derivative filtering and wavelet transform with log-Gabor-filters among these transforms. They are followed by fitting a GGD to obtain features in the same way as in BRISQUE. For the calculation of the image quality index the similarity method described in 4d in Section 4.1.2 is used with fitting a multivariate Gaussian distribution. The authors concluded that the features obtained after applying log-Gabor filters are the most influential on the quality. IL-NIQE method, according to the authors, is more computationally expensive than BRISQUE. In [114] other NSS IQA methods are compared in terms of time complexity and all other NSS methods mentioned in [65] are slower than BRISQUE. On the other hand, we can incorporate only part of the suggested features in order to avoid long computation time.

Other opinion-aware NSS methods discussed in [65], BIQI (Blind Image Quality Indices) and BLIINDS-II (BLind Image Integrity Notator using DCT Statistics), are several years older than the ones discussed in the previous paragraph. BIQI is introduced in [72]. Firstly, it determines distortions in the image by applying a wavelet transform, fitting a GGD, and applying SVM to the approximated parameters. After that, the same features are used in the SVR for these specific distortions and the results are combined proportionally to the probability of the distortions. In BLIINDS-II, discussed in [91], DCT is applied to the regions of the image and then GGDs are fitted to them. Several functions are applied to the approximated parameters of these distributions to obtain the features of the image. Then, the features are combined with a subjective score. Finally, a Bayesian approach is used to obtain quality scores assuming a multivariate GGD for these combinations.

There are two opinion-aware non-NSS methods mentioned in [65]: both [33] and [63] rely on image entropy. In [63] spatial and spectral local entropy values are calculated for 3 different frequency scales of an image. Spatial entropy is calculated directly from pixels values and spectral entropy — after application of DCT. These entropy values are used to construct features by pooling and averaging. Finally, the features are used in SVM. In [33] 1D pseudo-Wigner transform is used in several directions of image and the average or the standard deviation of normalized anisotropy values are used as quality metrics.

One more opinion-aware method is proposed in [38]. There, the authors use the idea of free energy to construct features. The authors down-sample an image and apply different Gaussian filters to it. Afterwards, the features are constructed using linear regression for free energy on some rational functions of means, variances, and covariances of the input and distorted images. Some other features are constructed using an image predicted by AR-model fitted to the input image. The last group of features is obtained in the same way as in BRISQUE in [70] by means of fitting a GGD. After that, SVR is applied to the features.

The next considered IQA method is a non-NSS opinion-unaware method of quality-aware clustering (QAC) which is proposed in [108]. In this paper, the training set of original images is distorted. After that, each image is split into overlapping regions (so-called patches) and for each patch similarity score between distorted and non-distorted patches is calculated. Then, the patches are clustered into so-called “quality clusters” based on the averaged and normalized similarity scores. Afterwards, 3 different Gaussian filters are applied to patches and the flattened outcomes are used as feature vectors for k-means clustering inside of each of the quality clusters. To estimate the quality of a new image the following procedure is performed. Firstly, it is split into patches and the feature vector for each patch is calculated. Then, for each patch the respective cluster and its’ centroid are determined. A weighted average of these centroids is considered as a quality estimate. This way any image can be classified as a weighted average of its' local qualities.

In an opinion-aware method in [95] the features are calculated after applying Daubechies 9/7 wavelet transform. Then, linear regression on these features is used in order to obtain a quality score.

In [36] several different types of features mentioned before are collected in order to train IQA model. The features are obtained in several different ways. Log-Gabor transform is applied and phase congruence (PC) entropy is calculated. Local contrast features are calculated based on the Gaussian-filtered image. Sharpness-related features are evaluated after applying wavelet transform. Brightness features are entropy values of brightness transformed images. Color features are saturation and a function of the mean and variances of color channels. Naturalness features are constructed as parameters of fitted Laplacian onto the normalized image and the mean of the darkest color of an image. Then SVR is applied to the features in order to assess the images based on scores obtained from comparing original and enhanced ones.

In [97] a random tree classificator is used for quality assessment. It uses technical features of images some of which are extracted based on the similarity between an enhanced and the original images. Furthermore, there are simplified versions of spatial envelopes used as features that were initially introduced in [76]. They represent the dominant spatial structure of an image. To get these features, firstly, DFT is used. Then, in order to get features from the transformed image, dimensionality reduction methods such as Karhunen-Loeve transform and PCA are suggested. Although it was applied in [76] to scene images, for example, for distinguishing man-made structures and nature, it can be still used to assess orientation or other features of documents.

To sum up this part of the literature review, many IQA methods exist and most of them combine different image features into a single quality metric. The assessment process sometimes is computationally expensive, hence, instead of the quality metrics proposed we can use only part of the features which are used in papers to construct a quality metric.

4.2 OCR methods and pre-processing

Features of images that influence the precision of OCR algorithm can be also found in research related to OCR itself and image pre-processing for it. Thus, it is important to understand how state-of-the-art algorithms work. Therefore, in this section, we review works on OCR algorithms and pre-processing techniques.

In [111] an extensive survey of existing OCR techniques up to 2014 is provided with examples of works that use different techniques: from traditional computer vision to deep learning algorithms. The main steps of discussed algorithms are text localization, text and character segmentation, and word recognition which can be performed in different ways.

For text localization methods color, edge and gradient features, and texture features (extracted by means of spectral transforms) and their combinations are usually used. Furthermore, dense neural networks or machine learning algorithms (such as SVM) can be applied to them in order to verify text regions. After the discovery of text regions, they are binarized: transformed into binary images and lines of text and characters are segmented sequentially. This is followed by the character recognition step and word recognition step which uses language knowledge in order to connect characters.

Since 2014, some progress was made in the area of OCR in each step. For example, in [116] a new method for text detection that is fast and accurate is introduced. An LSTM network is used in a new version of the character recognition algorithm tesseract as mentioned in the Chapter 3. New binarization methods are suggested: for example, improved existing methods as in [40] or new methods as in [39], which can be sometimes computationally expensive.

Sometimes, pre-processing of an image is performed before OCR itself. In a book about OCR [13] noise reduction, different types of normalization and compression are suggested as pre-processing techniques for OCR. However, it is also mentioned that pre-processing should be handled with care as it can result in worsening OCR.

As our problem covers only photos of invoices, we can also look into methods related to documents as these problems have some specific restrictions. For example, usually in documents, there is black text present with a clearly distinguishable background (white paper). If there is still variation in color, color-reduction (as in [74]) or conversion to grayscale algorithms can be applied. A survey with a comparison of these algorithms is presented in [49]. Among other problems that we can encounter are blurring, uneven or not sufficient illumination, document skew in a photo, cropped text, and damages or transformations of the paper (e.g., folds, fractures, or stains). Therefore, we look for the approaches which are suggested to correct for these problems and are not computationally expensive.

In [62] it is suggested to apply projective geometry methods in order to rectify documents and transform them into the scan-looking paper before application of OCR. However, one of the limitations of the suggested method is that a page should consist of dense text while invoices are usually sparse in text.

Furthermore, many papers suggest applying enhancement techniques aimed at improving different image characteristics such as sharpness or contrast before doing OCR. This way, for example, in [20] authors propose an image deblurring method based on deconvolution aimed specifically at text images. Learning-based methods are also suggested in the literature for cleaning images and enhancing resolution, such as in [11] or in [107] with the use of GANs.

Some papers approach several document image quality problems. For instance, in [61] the authors approach both rectification and correction for uneven illumination with the use of convolutional networks before applying OCR. And in [17] multi-plain segmentation approach is suggested for text regions allocation which allows correcting for uneven illumination.

Thus, we can see that OCR methods have shifted from computer vision approaches to mostly deep learning ones. However, some steps of OCR which do not require learning can be performed for feature extraction. This way, the text localization step will be further discussed in Section 4.3.2. Besides this, the discussed pre-processing methods for OCR can be also used as enhancement methods in this project.

4.3 Text images features

In this section, we will look into features that can be extracted specifically from document images. We will look into text skew and orientation detection methods in Section 4.3.1 and into text localization methods in Section 4.3.2.

4.3.1 Text skew and orientation detection

Text skew and orientation in images are useful features in IQA of document images. Skew detection can be performed both on a whole image and locally to detect skew in different regions of the image. This is useful as paper can be cramped or lie on an uneven surface so that different skew angles are observed in different parts of the image. Images are usually binarized and downscaled before detecting skew.

Most of the methods used for skew detection can be classified into the methods which use the Hough transform, interline cross-correlation, connected component analysis, and project histograms described in [43], [109], [79] and [23] respectively.

In [43] the Hough transform is applied to a binarized image. It converts the image from Cartesian (x, y) space to (ρ, θ) space by applying the formula 4.5 to black pixels and adding 1 to the respective (ρ, θ) coordinates.

$$\rho = x\cos\theta + y\sin\theta \tag{4.5}$$

For every black pixel (x, y) ρ is calculated for all values of $\theta \in [0°, \ldots, 180°]$. This range is taken smaller in practice as we do not expect large skew angles. In [43], before application of the Hough transform the authors downscale the image and apply a so-called bursting procedure vertically in order to increase speed. During this procedure, the sequential black pixels are substituted by only one gray pixel at the end of the sequence with the intensity equal to the number of black pixels in this sequence. As now there are gray pixels, the value in respective (ρ, θ) coordinates is increased by the intensity value. θ coordinate of the largest value is taken as text orientation.

In [109] interline cross-correlation is calculated for lines in an image and then accumulated in order to estimate the skew of the document. Thus, for reliable estimates, this method requires a lot of text in the image which is not the case in our project. In [23] the region with the maximum density of black pixels per row is rotated by different angles and the horizontal projection histogram for each of them is evaluated. The rotation angle for which the mean square deviation is maximized is taken to be the skew angle.

Several skew detection methods similar to described above are proposed in [58]. There, the authors also split an image into regions and then propose 7 criteria for choosing text-dominated regions for analysis. Then, they suggest using the projection profiling method or the Hough transform for skew and orientation detection.

The method suggested in [55] uses profile analysis for skew detection. The image is split into regions and for each of them a so-called complexity variance $V(\theta)$ is calculated for different rotation angles θ of the image. Complexity variance $V(\theta)$ is defined as the variance of the number of transitions between black and white pixels in a line. The value of θ for which $V(\theta)$ is maximum is considered as a skew angle of a region. In this paper, it is required to choose regions with a sufficient amount of text which might be challenging for images with sparse text.

In [79] the authors suggest a connected components method that corrects for noise and improves on the speed of algorithms that use the Hough transform. Firstly, connected components are discovered. In the documents, connected components usually represent letters. Then, too large and too small components are discharged and two new images are constructed by retaining only top-left and only bottom-right points of the component bounding boxes. Afterwards, pixels in both new images are clustered in lines and the average of slopes of each line is considered as a skew metric. A similar method is proposed in [14]. Besides this, connected components can also be used in projection profiling which is applied not to an original image but to the image where only the centroids of connected components are kept as described in [10]. Another skew detection method based on connected components analysis is introduced in [45]. There, the authors suggest a method to identify eligible connected components and to calculate skew angle options based on them. Then, the overall skew angle can be chosen by applying the projection profile method or Dixon's Q outliers test.

The authors mention that the methods described above require a lot of text on the image which is not always the case for images of invoices. Thus, we also study the research related to skew detection in the images with sparse text.

One of such methods is proposed in [94]. Firstly, an image is downscaled to make text resemble a texture and then apply methods for detecting texture skew. For this, the image is convolved with two Gaussian filters in order to obtain two new images, which are considered as gradients. Then, the magnitude and direction of the gradient are calculated. Finally, the dominant direction is calculated for different windows of the image by applying Rao's [89] or Chaudhuri's [15] formula. The windows were chosen in such a way that they include some text and are not completely blank. The use of Chaudhuri's formula is slower according to the authors, therefore, in their experiments, only Rao's function was used.

In [85] the DFT is applied to the image and the direction of the highest density is considered as a skew angle. In [44] the authors propose to apply a mask to the Fourier transformed image before a search for direction in order to remove a bias at 45°.

In [25] authors propose to use morphological operations for skew detection. They are used in order to convert text on the image into black bands and to remove noise from the image. Then, only baseline pixels of these bands are retained and component labelling is performed in order to detect lines. Finally, the median skew angle of these lines is reported as a skew angle. The authors also propose several speed-ups of the morphological operations.

Most of the algorithms described above do not approach the problem of curved strings: they can only detect skew. In [19] the authors propose an algorithm that will also detect the curving structure of a text. However, text regions should be chosen manually or by another algorithm beforehand. Then, connected components (letters) are grouped into strings following several proposed rules. Finally, the orientation of the strings is evaluated by means of morphological operations as described in the method from [18].

Thus, in this section, we have discussed various document skew and orientation detection and evaluation methods. Document skew is an often problem that emerges during OCR. Therefore, it is worth evaluating skew angles and using them as features in our project.

4.3.2 Text regions localization

Let's look more thoroughly into text localization methods. This problem is researched a lot as it is a part of OCR methods. Modern methods propose to use convolutional neural networks (CNN) in order to locate the text regions ([100]). As these networks do not require training on our specific images, it is possible to train such a network on some images corpus and then apply it to our data.

However, let's first look into not deep learning methods. Most of them can be split into 3 categories according to [115]: texture-based, connected components (CC) based, and edge-based methods. The texture-based methods treat text as a specific type of texture and, depending on the features, distinguishes it from the background. The connected components method detects connected components and then merges the large ones together. Finally, the components related to the text are distinguished from background components by geometric features. The last group of methods detects regions of dense edges. However, these methods are bad at differentiating texture and text. Thus, they might work only if the background on the images is absent (e.g., only a piece of paper is present on the image).

Texture-based methods usually require some transforms of an image and a machine learning algorithm in order to distinguish the text using features. This way, in [112] the authors propose to use a wavelet transform in order to extract features, and SVM to classify text and non-text regions. In [16] the authors use AdaBoost learning with features such as the entropy or the mean and standard deviation of intensity in the blocks of gradient images. Thus, if we would like to avoid any learning, we should look into CC based methods.

One of such methods is proposed in [115]. There, the authors suggest detecting corner points and constructing a binary image from them (with white pixels representing corners). Then, they apply a morphological dilation and, finally, using features of the corner points regions (area and ratio of white pixels in the bounding box, orientation and aspect ratio of white regions), they distinguish text from non-text regions.

Another CC based method is suggested in [29]. There, the authors use stroke width transform (SWT) which substitutes each pixel of an image with the estimated width of the stroke (straight line of which a letter consists) in which it is contained. Afterwards, connected components are detected based on the transformed image. The components with the large variance, too small and too large ones, and the ones that contain others inside are discarded. The remaining components are assumed to be letters. Finally, the letters are clustered into text lines taking into account stroke width, the distance between them, and the height of the components. The lines which are too short are also discarded as noise.

The discussed text localization methods can be used to produce new features in this project such as, for example, distance from the text to the border of the image.

4.4 Image enhancement techniques

The second part of the project covers image enhancement. Thus, we have to review the literature on this topic, too. We have already mentioned some methods in Section 4.2 which are used as preprocessing techniques for OCR. In this section we will focus both on general image enhancement techniques (Section 4.4.1) and images enhancement techniques aimed at document images (Section 4.4.2). Besides this, we will look into text skew correction methods in Section 4.4.3. As there is extensive research done in the image enhancement domain, we will omit methods with high computational complexity.

4.4.1 General image enhancement techniques

Let's start with classification of general image enhancement techniques. Most of them are applied to grayscaled images. These methods can be classified into spatial and frequency domain techniques (as mentioned in [90], [104], [47], and [68]).

Spatial methods

Various spatial methods are suggested in the enhancement techniques surveys [104] and [68]. In [68] the focus is made on spatial domain methods. In further classification in [68] the authors split spatial methods into point-processing operations and histogram operations. Among the point-processing methods, they mention taking negative of an image, thresholding, logarithmic transformation of the image, gamma-transformation (exponentiation of the image into power γ), piece-wise linear transformation, and gray-scale slicing (band-pass filter in the spatial domain). The histogram methods mentioned in the paper are histogram equalization, histogram matching to another image, and local histogram equalization.

The histogram equalization method can be found in [47]. In [84] the authors propose an adaptive histogram equalization method for contrast improvement which does not introduce too much noise as the previous histogram equalization approaches did and which is computationally less complex. In this method, the grid is introduced and for each grid pixel, its intensity is set proportionally to the pixel rank among the pixels surrounding it. Then, all other pixel values are interpolated from the surrounding grid pixel values.

An improved gamma-transformation method, a so-called adaptive gamma correction, is suggested in [87] and later improved in [12]. In this method, power γ is calculated using a normalized histogram of the image and differs for different values of pixels.

Slightly different split of spatial domain methods is presented in [90] and [104]. The following 3 groups of methods are proposed there: smoothing, gray-level scaling, and edge enhancement methods. The smoothing methods are mostly implemented as weighted matrix filters. These methods help in reducing noise while blurring the image. The gray-level scaling methods are analogous to the histogram methods mentioned in [68]. The edge enhancement methods are presented as rational formulas applied to the gradients and original image to improve the sharpness of details. For example, subtracting the multiple of the Laplacian of the image as described in [90]. Moreover, edges can be enhanced by applying a high-pass filter to the image as the edges are high-frequency components of the image.

Many papers on spatial enhancement methods focus on improving image quality in certain aspects. More specifically, the authors mainly suggest methods for increasing sharpness, increasing contrast, or reducing noise.

In [34] the authors use the idea of representing an image as a Laplacian pyramid in order to increase sharpness. The idea is that the image can be decomposed into a high-frequency component L_0 with emphasized edges and a low-frequency component by applying the respective filters. Afterwards, the low-frequency component can be again decomposed into a high-frequency L_1 and a low-frequency one and so on. The enhancement method is based on predicting L_{-1} — a previous high-frequency component. For this, L_0 is first extracted and, then, L_{-1} predicted as in eq. 4.6.

$$L_{-1} = \mathcal{H}(s \times BOUND(L_0)), \tag{4.6}$$

where $\mathcal{H}$ is a high-pass filter represented as 5×5 kernel and $BOUND$ is a function defined in eq. 4.7 which allows to avoid extreme values of pixels.

$$BOUND(x) = \begin{cases} T, & \text{if } x > T, \\ x, & \text{if } -T \leq x \leq T, \\ -T & \text{if } x < -T \end{cases}, \tag{4.7}$$

where $T = (1 - c)L_{0max}$ with L_{0max} being the maximum values of L_0.

The constants s and c can be estimated theoretically based on the filter variance. The enhanced image is then constructed as the sum of the input image and L_{-1} which has emphasized edges.

For noise improvement, window filtering techniques are often used. This way, in [53], a center-weighted median filter is proposed for noise reduction. There, each pixel is substituted by the median of weighted values inside a window surrounding this pixel.

Separately applied improvement techniques for denoising and sharpening usually do not give a good result. The method applied last distorts the enhancement by the previous method and can introduce artifacts. Thus, many authors approach the problem of sharpening and denoising the image simultaneously. This way, in [75] the authors propose a method that improves images both in terms of sharpness and noise. The noise filtering algorithm consists of 4 stages: for each pixel, the neighbourhood is selected, then the pixels to be used for enhancement are selected in this neighbourhood. Afterwards, the weights are assigned to these pixels and the weighted sum is normalized. In parallel, a high-pass filter in form of a 3×3 filter is applied to the original image and, then, both results are combined with some pre-defined coefficients in order to get an improved image.

Another method that corrects both for noise and blur is proposed in [50]. There, the authors suggest using an optimal unsharp mask to improve image sharpness and reduce the noise level. In this paper, the enhanced image $\hat{f}$ is constructed from the original image g using the formula in eq. 4.8.

$$\hat{f}(m,n) = g(m,n) + \lambda(g(m,n))\mathcal{H}(g(m,n)), \tag{4.8}$$

where $\mathcal{H}$ is a high-pass filter and λ is a coefficient. This algorithm requires training on the set of images close to ideal to learn coefficients λ for different pixel values. In this method, we again sum up the original image and an image with emphasized edges as we pass the original through a high-pass filter.

One more training-based method built on unsharp masks is proposed in [113]. There, the authors use an adaptive bilateral filter (ABF). It is an improvement of a bilateral filter and in addition to softening the noise, it sharpens the edges. One more similar method is proposed in [83]. It is called adaptive guidance filtering (AGF) and differs from ABF by the weights which are used in the filter.

In [7] authors propose an approach that combines a smoothing method (sigma filtering ([60]) is chosen by the authors due to its simplicity) and unsharp masks: a so-called constrained unsharp masking method. Smoothed image and image, passed through a high-pass filter are summed up with pre-defined coefficients, and the resulting image is clipped (by applying a function analogically to $BOUND$ in eq. 4.7).

Sigma filtering introduced in [60] is a denoising method. In this method each pixel $x(i,j)$ of the image is substituted by the average value of pixels lying in the specified window around (i,j), and specified pixel values range around $x(i,j)$.

In [77] the authors suggest shock filters to improve image sharpness in presence of noise. In this paper, a blurred image is considered as a convolution of an image of high quality. Thus, for deconvolution, a partial differential equation with an improved image as an unknown function is constructed and then solved by numerical methods. According to the authors, this method allows saving total variance, so that the edges and other lines on the image are not lost.

Another method relying on differential equations is suggested in [82]. The authors use there the anisotropic diffusion for edges enhancement. In this paper, an iterative numeric method for solving partial differential equations is used on the grid of an image.

The contrast of the image is another feature to be enhanced. For example, in [88] the retinex function is used to enhance image contrast. In this paper, the authors also propose to use it for color restoration. Another method is proposed in [48]. There, the image is passed through a low-pass filter, then, the detailed and smooth regions of the image are separated and, finally, contrast gain is added to detailed regions according to some thresholds.

To sum up, there is a wide range of available spatial image enhancement methods that are aimed at different image deteriorations. In this project, we implement some of them and investigate their performance on document images.

Spectral methods

The survey of spectral methods can be found in [5]. Generally speaking, they suggest the following algorithm for all spectral domain methods:

1. Apply a spectral transform to an image $X(m,n)$;
2. Apply a filter to the transformed image $X'(m,n)$;
3. Apply the inverse spectral transform to the filtered image $f(X'(m,n))$.

Among the suggested spectral transforms there are DFT, real DFT, and DCT of 2 types. The main purpose of the filtering methods is to reduce large transform coefficients which correspond to low frequencies and make them closer to small ones which correspond to high frequencies (and, hence, edges). Thus, after the inverse spectral transform, the image with strengthened edges will be obtained. The authors of [5] propose such filtering methods as multiplication by a matrix filter, alpha-rooting, modified unsharp masking, and a filter motivated by the human visual system (HVS). Alpha-rooting is presented in eq. 4.9.

$$f_\alpha(X'(m,n)) = X'(m,n)|X'(m,n)|^{\alpha-1}. \tag{4.9}$$

Modified version of unsharp masking is defined as in eq. 4.10.

$$f_{\text{UM}}(X'(m,n)) = X'(m,n)(\mathcal{H}_L(X'(m,n))(1-C)+C)\mathcal{H}_N(X'(m,n)). \tag{4.10}$$

where $\mathcal{H}_L$ and $\mathcal{H}_N$ are two different low-pass filters and C is some constant.

Finally, the HVS filter is a non-linear filter that amplifies mid-frequencies to which the human eye is most sensitive. Thus, it is not very relevant for the project as we focus on improvements for computer vision instead of human vision. The authors report that both types of DCT with the modified unsharp mask or the HVS filter provide the best result.

In [4] the authors elaborate more on spectral methods. The proposed approaches follow the same structure with the difference that instead of filtering they suggest applying some function $O(x,y)$ to the magnitude of the transformed image. In their research, the authors try different spectral transforms with 4 choices of $O(m,n)$. They are constant, modified alpha-rooting, logarithmic, and a product of modified alpha-rooting and logarithmic functions, so-called log-alpha rooting. It is given in eq. 4.11.

$$f_{\log\alpha}(X'(m,n)) = \log(|X'(m,n)|^\gamma + 1)^\beta X'(m,n)|X'(m,n)|^{\alpha-1}. \tag{4.11}$$

They also introduce the performance metric and according to it, the best results are obtained with Walsh transform. The best choice of function $O(m,n)$ might vary depending on the images. The authors also mention the possibility of application of different $O(x,y)$ to different regions of the image.

Thus, spectral enhancement methods have a similar structure. The differences are only observed in the spectral transforms and filtering functions which are applied to the transformed images. In this project, we implement different spectral transforms, filtering functions, and compare the results.

4.4.2 Image enhancement for text images

We have discussed the enhancement methods that can be applied to any type of image. However, it is usual that natural scene sharpness enhancement methods do not perform well on text images according to [20]. That is why we also discuss the research in the area of enhancement methods which can be applied specifically to the text images.

Recently, many text image enhancement methods have been proposed based on kernel estimation methods. The blurred image b can be represented as $k * l + n$, where k is a blurring filter, also referred to as kernel, l is a latent (not blurred) image and n is noise. The methods are based on estimating the blurring filter k and performing deconvolution to obtain the unblurred image.

In [20] the authors propose a method for text deblurring. They suggest solving an optimization problem in eq. 4.12.

$$\arg\min_{l,k,a} ||b - k * l||^2 + \rho_l(l) + \rho_k(k) + \rho_a(a) + \beta||l-a||^2, \tag{4.12}$$

where $\rho_l(l)$ and $\rho_k(k)$ are priors for the latent image and the blurring filter respectively, a is an auxiliary image which enforces domain-specific properties, $\rho_a(a)$ is a cost function, and β is a weight of regularization term. Then, they suggest solving this optimization problem iteratively: assuming we know k we start with solving it for l with initial value $a = l$, then solve it for a using this l and repeat the process increasing β. Each step involves calculations of several FFT. Afterwards, k is estimated and the whole process is repeated again with the estimated k. In the end, deconvolution is performed using k, a and l.

The authors report the computational time of a few minutes for an image of size 425×313 on a PC running MS Windows 7 64bit version with Intel Core i7 CPU and 12GB RAM. This is too slow for our project as we are looking for the fast methods which will take maximum 2-3 seconds.

A simpler method was proposed in [80] with use of L_0 regularization and gradient priors. The optimization problem posed in this paper is similar to eq. 4.12 and given in eq. 4.13.

$$\arg\min_{l,k} ||b - k * l||^2 + \gamma ||k||^2 + \lambda(\sigma P_t(l) + P_t(\nabla l)), \tag{4.13}$$

where γ, λ and σ are weights, ∇l is a gradient of the image and $P_t(x) = ||x||_0$ is a prior, where $||x||_0$ counts the number of 0's in x. The authors also suggest solving this minimization problem iteratively by introducing an auxiliary variables and solving 3 alternative minimization problems. Each step, as in the previous method, involves computation of FFT.

This method is reported to be less computationally expensive, although, it still requires the computational time of 50 seconds for an image of size 255×255 on a desktop computer with an Intel Xeon processor and 12 GB RAM.

A faster deblurring method is proposed in [28]. Its computational time is 8.7 seconds on an image of a size 255×255 (on a PC with an Intel i7 CPU and 8GB memory) according to the authors which is still unaffordably long. It is reported to perform well both on text images and natural scene images. The optimization problem they offer to solve is given in eq. 4.14.

$$\arg\min_{k,l} ||b - k * l||^2 + \alpha \frac{D}{D^2 + \epsilon} + \gamma ||k||^2, \tag{4.14}$$

where α and γ are weights, ϵ is a smoothing-enhancement parameter and D is the magnitude of the gradient of b. As well as before, the authors suggest solving this optimization problem by solving iteratively alternative optimization problems using FFT and fixed point iteration algorithm.

However, all these methods are computationally expensive and, hence, are not useful in the problem approached in the current project. Thus, we will not use them for image enhancement.

4.4.3 Skew correction techniques

Most of the papers on skew detection suggest a simple method for skew correction: rotate an image by the detected angle. However, the images can have a different skew in different regions and, so-called slant, which is an angle of the characters tilt. Slant was not observed as a problem in this project. However, slant correction methods can be applied to regions of the image in order to rotate them by the correct angle.

For example, in [99] the authors suggest a slant correction by shear operation. For each pixel (i, j) its new coordinates (x, y) are calculated as in eq. 4.15.

$$\begin{cases} y = j \\ x = i - (height - j)\tan\theta \end{cases}, \tag{4.15}$$

where θ is a slant angle and *height* is the text height.

There is also research made on the local skew correction. For example, in [93] the authors suggest using a connected components approach to detect characters and after that group them into the words. Afterwards, the lines and regions of text with similar slope are formed. Eventually, the regions are rotated with a help of bilinear interpolation to avoid the overlapping of text areas.

In this project, we will focus on the simple skew correction methods when the image is rotated by the angle discovered during the skew detection step.

Chapter 5

Data preparation and labelling

After reviewing the literature on image features and enhancement methods, we should label the data for classification. For this, the ORTEC algorithm was run on all available images and the outcomes were saved. We want our algorithm to focus on evaluating the quality of an image of an invoice and not the content of it. Thus, we cannot simply use the outcome of the ORTEC algorithm as labels for the data as it also analyzes the content of the invoice. As mentioned in Chapter 3, we differentiate between correctly and incorrectly rejected images and between correctly and incorrectly partially recognized images. Therefore, we have to go through all the invoices manually to distinguish these cases. Then, we can introduce a new labelling that suits the problem at hand.

We introduce a labelling of 3 classes: positive, partial and negative. When splitting the images, we were considering that after classifying an image as negative a user of the application will be asked to take a new photo. Thus, in our labelling, we want to have a negative label on the images whose recognition can be improved by taking a new photo. The images are split into the classes according to the following, previously discussed in Chapter 3, possibilities:

- Positive class:
  - Fully recognized;
  - Correctly partially recognized;
  - Correctly rejected;
- Partial class:
  - Incorrectly partially recognized;
- Negative class:
  - Not recognized (the same as incorrectly rejected);
  - Cropped images: the images where part of the text important for recognition is not seen.

After manual labelling of the data, we get 312 positive examples, 103 partial examples, and 114 negative examples.

Later, in Section 8.1, we will consider classification into 2 classes, for which we will have to include partial class either into the positive or into the negative class.

Furthermore, some new data was generated. It was made to increase the number of negative and partial examples with specific image quality problems. For example, invoices rotated by different angles, cramped invoices, or invoices out of focus. Firstly, several scans of invoices were printed and new photos of them were made with different illumination, orientation, and paper folding. This provided with another 46 negative examples and 9 partial examples which were immediately included in the dataset. Thus, for the analysis, we have 584 images: 312 positive, 112 partial, and 160 negative examples.

Besides this, some data was generated by blurring recognized images by means of Gaussian blur kernel filter with different sizes and variances. This data is used only in Section 8.2.7 about dataset enlargement to determine the impact of expanding the dataset with such kind of data. As the blurred images may be very similar to original ones in terms of all the features except sharpness, we should handle this data with care.

Besides dataset labelling and expansion, we should also make an adjustment to the ORTEC algorithm as all invoices used in the project are anonymized as mentioned in Chapter 2. Thus, we have to exclude the client's details from the mandatory fields when running the ORTEC algorithm.

To sum up, we have introduced a labelling that suits the classification problem that we are solving. Furthermore, we have generated some new images in order to expand the dataset. Finally, we adjusted the ORTEC algorithm so that it can handle anonymized data.

Chapter 6

Approaches

In this chapter, we discuss the possible approaches to image classification and image enhancement problems.

We start by considering possible approaches to classification. Firstly, we will look at the problem of classification of invoices into 3 classes: positive, partial, and negative. The first idea is to use a convolutional neural network (probably, pre-trained on some other data) on the raw images of invoices. However, due to the insufficient amount of available training data, it might be not a good idea. On the other hand, the dataset can be extended by doing further data augmentation from available images and by manually creating fake invoices and taking photos of them. However, data augmentation by transforming the existing invoices does not provide completely new data. Furthermore, faking a sufficient number of invoices is a time-consuming process. Thus, we would better consider other classification methods.

That is why our primary approach is to extract features from the images, such as colors, brightness, and contrast among others in order to run one of the machine learning algorithms (e.g. k-NN, decision trees, or SVM). Possible choices for the features can be taken from image statistics, IQA, and text features papers discussed in Sections 4.1 and 4.3, and from preprocessing section of the OCR papers discussed in Section 4.2. When the desired features are extracted we can classify images based on them and look into incorrectly classified ones in order to choose new features that might be able to capture their specifics.

Furthermore, classification into 2 classes instead of 3 classes can be investigated. Then, we have to include the partial class into the positive or negative class. Both approaches might be suitable: a partial example can be considered a positive as the partial example is analyzed by the ORTEC algorithm and a user does not have to take a new photo, he has only to add missing details. On the other hand, it might worth including it into the negative class as the reason for being partially recognized might be the bad quality of the image. We will approach this question in Section 8.1.

In the beginning, the complexity of the developed algorithm would not be considered as an important factor and we will be looking mostly at the evaluation metrics. After investigating the quality of suggested options, we will look into which of them performs better in terms of both complexity and quality. Moreover, images can be downscaled before running the algorithm which will improve the speed of it. Thus, we can use downscaling to balance performance and speed, too.

Next, let's consider several approaches to the image enhancement problem. The features extracted for classification can also provide us with the information on which image characteristics are contributing the most to the recognition of invoices. Thus, the improvement of image quality might be focused on these features.

Furthermore, after the application of enhancement techniques, some images might become recognized by the ORTEC algorithm and some might become not recognized. Thus, we can classify the images according to the enhancement techniques that help to improve their quality. Therefore, we would be able to apply the enhancement technique only to the images on which we assume it to have a positive impact.

Thus, we have discussed the approaches that we will use for solving the classification problem and enhancement problem. In the next sections, we will elaborate on these methods.

Chapter 7

Features extraction

In this chapter, we discuss in detail the implemented methods for feature extraction introduced in Section 4.1. We choose several methods for each of the image characteristics, such as brightness, contrast, and sharpness. We choose the methods which are simple in implementation, have relatively low computational complexity, and perform well in evaluating the respective image characteristics according to the authors of the papers.

After implementing these methods for each of the image characteristics, we implement the extraction of the features used in the state-of-the-art general IQA methods BRISQUE and IL-NIQE. Finally, we implement the extraction of relevant document-related features such as, for instance, skew.

7.1 Image characteristics

In this section, we discuss the implemented methods for the extraction of such features as brightness, contrast, and sharpness as well as some other image characteristics.

To begin with, we extract basic image features such as width and height of an image in pixels, size of the image in bytes, and the average values of the 3 RGB (red, green, and blue) color channels.

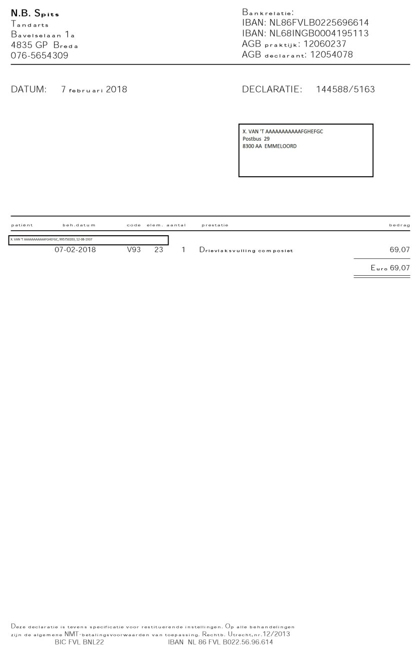
Afterwards, we extract brightness features, described in [6]. There are several methods suggested which use RGB color channels values, such as the average (eq. 7.1), the maximum (eq.7.2), luma (eq. 7.3) and luminance (eq. 7.4).

$$B_{\text{av}} = \frac{R+G+B}{3}. \tag{7.1}$$

$$B_{\max} = \max(R, G, B). \tag{7.2}$$

$$B_{\text{luma}} = 0.299R + 0.587G + 0.114B. \tag{7.3}$$

$$B_{\text{luminance}} = 0.2126R + 0.7152G + 0.0722B. \tag{7.4}$$



N.B. Spits
Tandarts
Bavelselaan 1a
4835 GP Breda
076-5654309

Bankrelatie:
IBAN: NL86FVLB0225696614
IBAN: NL68INGB0004195113
AGB praktijk: 12060237
AGB declarant: 12054078

DATUM: 7 februari 2018

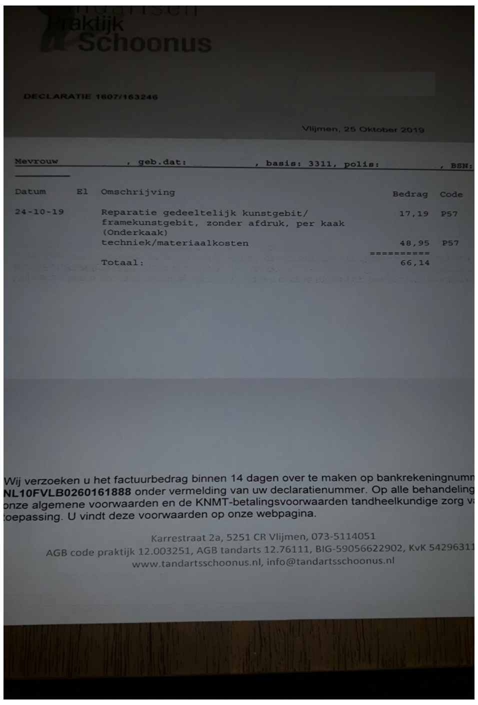
DECLARATIE: 144588/5163

X. VAN 'T AAAAAAAAAAAFGHEFGC
Postbus 29
8300 AA EMMELOORD

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
X. VAN 'T AAAAAAAAAAAFGHEFGC, 995750203, 12-08-1937						
	07-02-2018	V93	23	1	Drievlaksvulling composiet	69,07
						Euro 69,07

Deze declaratie is tevens specificatie voor restituerende instellingen. Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing. Rechtb. Utrecht,nr.12/2013
BIC FVL BNL22 IBAN NL 86 FVL B022.56.96.614

(a) Example of an image with high brightness.



Praktijk
Schoonus

DECLARATIE 1607/163246

Vlijmen, 25 Oktober 2019

Mevrouw , geb.dat: , basis: 3311, polis: , BSN:

Datum	El	Omschrijving	Bedrag	Code
24-10-19		Reparatie gedeeltelijk kunstgebit/ framekunstgebit, zonder afdruk, per kaak (Onderkaak)	17,19	P57
		techniek/materiaalkosten	48,95	P57
		Totaal:	66,14	

Wij verzoeken u het factuurbedrag binnen 14 dagen over te maken op bankrekeningnumm
NL10FVLB0260161888 onder vermelding van uw declaratienummer. Op alle behandeling
onze algemene voorwaarden en de KNMT-betalingsvoorwaarden tandheelkundige zorg v
toepassing. U vindt deze voorwaarden op onze webpagina.

Karrestraat 2a, 5251 CR Vlijmen, 073-5114051
AGB code praktijk 12.003251, AGB tandarts 12.76111, BIG-59056622902, KvK 54296311
www.tandartsschoonus.nl, info@tandartsschoonus.nl

(b) Example of an image with low brightness.

Figure 7.1: Examples of images with high and low brightness.

All of these brightness metrics appear to be correlated. Thus, let's look at the examples for B_{av}. The largest values of B_{av} are obtained for scans (up to 250 as in Fig. 7.1a) and the lowest values — for dark images (as in Fig. 7.1b with brightness 84) as expected.

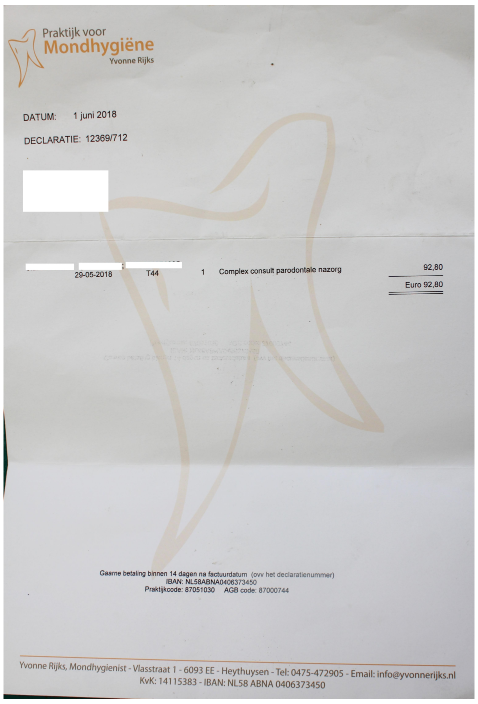
The next step is to extract contrast features. The following methods are considered: ratio (eq. 7.5), Michelson (eq. 7.6), and RMS (eq. 7.7). Here, B_i is a brightness metric and $i \in \{\text{av}, \text{max}, \text{luma}, \text{luminance}\}$.

$$C_{\text{ratio}} = \frac{\text{max}B_i - \text{min}B_i}{\text{av}B_i}. \tag{7.5}$$

$$C_{\text{Mich}} = \frac{\text{max}B_i - \text{min}B_i}{\text{max}B_i + \text{min}B_i}. \tag{7.6}$$

$$C_{\text{RMS}} = \text{std}(B_i). \tag{7.7}$$

Michelson contrast metric does not provide any information in our setting as, due to anonymization with white boxes used in many cases, the lowest brightness is 0 and the highest is 255 which results in the same Michelson contrast for many different images. Two other contrast metrics perform better. The ratio contrast metric returns low values mostly for scans while RMS also returns low values for images such as in Figure 7.2a (0.07 while the mean for the feature is 0.17). The images with high contrast, according to both ratio and RMS metric, are usually the ones with dark background as in Figure 7.2b (the image with the highest RMS



Praktijk voor
Mondhygiëne
Yvonne Rijks

DATUM: 1 juni 2018

DECLARATIE: 12369/712

[redacted]

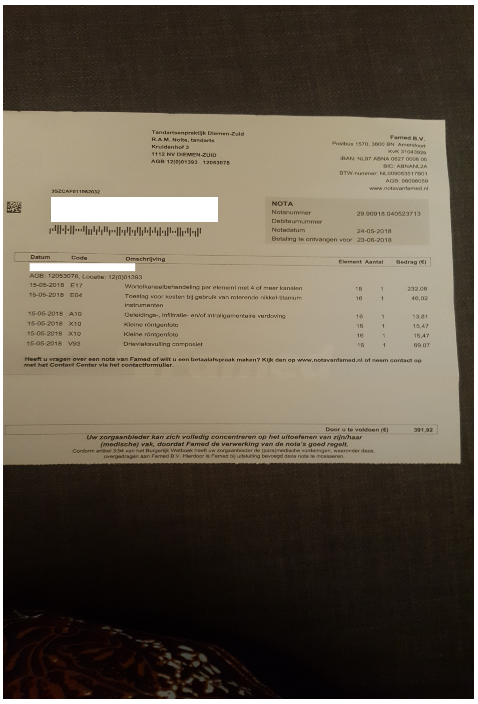
[redacted] [redacted] [redacted]

29-05-2018	T44	1	Complex consult parodontale nazorg	92,80
				Euro 92,80

Gaarne betaling binnen 14 dagen na factuurdatum (ovv het declaratienummer)
IBAN: NL58ABNA0406373450
Praktijkcode: 87051030 AGB code: 87000744

Yvonne Rijks, Mondhygienist - Vlasstraat 1 - 6093 EE - Heythuysen - Tel: 0475-472905 - Email: info@yvonnerijks.nl
KvK: 14115383 - IBAN: NL58 ABNA 0406373450

(a) Example of an image with low RMS contrast.



Tandartsenpraktijk Diemen-Zuid
R.A.M. Nolte, tandarts
Kruidenhof 3
1112 NV DIEMEN-ZUID
AGB 12(0)01393 12053078

Famed B.V.
Postbus 1570, 3800 BN Amersfoort
KvK 31043929
IBAN: NL97 ABNA 0627 0008 00
BIC: ABNANL2A
BTW-nummer: NL009053517B01
AGB: 98098059
www.notavanfamed.nl

3SZCAF011962032

[redacted]

NOTA

Notanummer	29.90918.040523713
Debiteurnummer	
Notadatum	24-05-2018
Betaling te ontvangen voor	23-06-2018

[redacted]

Datum	Code	Omschrijving	Element	Aantal	Bedrag (€)
AGB: 12053078, Locatie: 12(0)01393					
15-05-2018	E17	Wortelkanaalbehandeling per element met 4 of meer kanalen	16	1	232,08
15-05-2018	E04	Toeslag voor kosten bij gebruik van roterende nikkel-titanium instrumenten	16	1	46,02
15-05-2018	A10	Geleidings-, infiltratie- en/of intraligamentaire verdoving	16	1	13,81
15-05-2018	X10	Kleine röntgenfoto	16	1	15,47
15-05-2018	X10	Kleine röntgenfoto	16	1	15,47
15-05-2018	V93	Drievlaksvulling composiet	16	1	69,07

Heeft u vragen over een nota van Famed of wilt u een betaalafspraak maken? Kijk dan op www.notavanfamed.nl of neem contact op met het Contact Center via het contactformulier.

Door u te voldoen (€) 391,92

Uw zorgaanbieder kan zich volledig concentreren op het uitoefenen van zijn/haar (medische) vak, doordat Famed de verwerking van de nota's goed regelt.

Conform artikel 3:94 van het Burgerlijk Wetboek heeft uw zorgaanbieder de (para)medische vorderingen, waaronder deze, overgedragen aan Famed B.V. Hierdoor is Famed bij uitsluiting bevoegd deze nota te incasseren.

(b) Example of an image with high RMS and ratio contrasts.

Figure 7.2: Examples of images with high and low contrast according to the RMS and ratio contrast metrics.

contrast of 0.26). Thus, it does not give a lot of information on the contrast of the invoice itself in presence of a dark background.

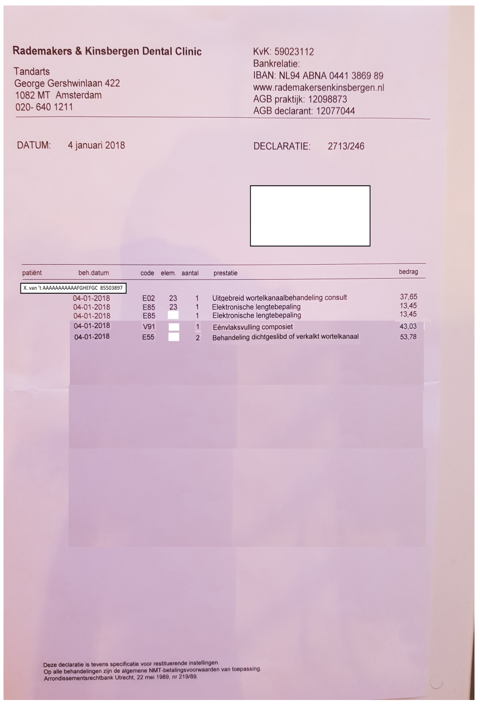
Most of the calculated contrast values also turn out to be correlated as they are calculated based on correlated brightness metrics. Thus, they are excluded from the set of features in the further analysis. This will be elaborated more in Section 8.2.3 regarding feature selection.

The next step is to include the average hue and saturation pixel values. For this, a Python function is used which converts an image from RGB color channels to HSV (hue, saturation, value). Value channel is equivalent in this case to eq. 7.2. Low hue and saturation levels are usually obtained on scans. In Figure 7.3 the examples of images with high hue level (Fig. 7.3a) of 153 (mean hue level is 44) and high saturation level (Fig. 7.3b) of 159 (mean saturation level is 33) are presented.

After that, several sharpness metrics are included in the feature set. The first considered metrics are so-called Laplace sharpness metrics, which are discussed in [81]. There, the image is firstly convolved with the Laplace filter l (eq. 7.8). Then, the mean and the standard deviation of the pixels after convolution are considered as the Laplace sharpness metrics.

$$l = \frac{1}{6}\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & 1 & 0 \end{pmatrix}. \tag{7.8}$$

Both the mean and the standard deviation Laplace sharpness have small values on blurred



Rademakers & Kinsbergen Dental Clinic

Tandarts
George Gershwinlaan 422
1082 MT Amsterdam
020- 640 1211

KvK: 59023112
Bankrelatie:
IBAN: NL94 ABNA 0441 3869 89
www.rademakersenkinsbergen.nl
AGB praktijk: 12098873
AGB declarant: 12077044

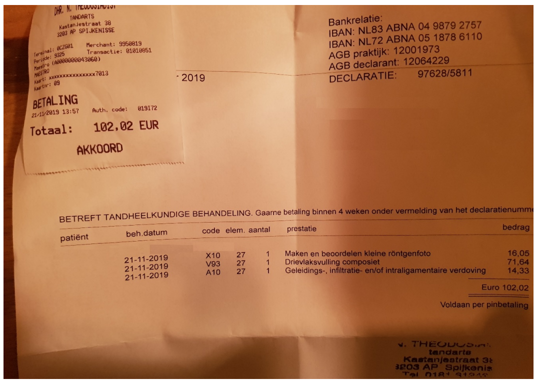
DATUM: 4 januari 2018

DECLARATIE: 2713/246

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
X. van 't AAAAAAAAAAAFGHEFGC 85503897						
	04-01-2018	E02	23	1	Uitgebreid wortelkanaalbehandeling consult	37,65
	04-01-2018	E85	23	1	Elektronische lengtebepaling	13,45
	04-01-2018	E85		1	Elektronische lengtebepaling	13,45
	04-01-2018	V91		1	Eénvlaksvulling composiet	43,03
	04-01-2018	E55		2	Behandeling dichtgeslibd of verkalkt wortelkanaal	53,78

Deze declaratie is tevens specificatie voor restituerende instellingen.
Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing.
Arrondissementsrechtbank Utrecht, 22 mei 1989, nr 219/89.

(a) Example of an image with high hue level.



DHR. N. THEODOSIADIS
TANDARTS
Kastanjestraat 38
3203 AP SPIJKENISSE

Terminal: 0CZG01 Merchant: 9950819
Periode: 9325 Transactie: 01010851
Maestro (A0000000043060)
MAESTRO
Kaart: xxxxxxxxxxxxxxx7013
Kaartnr: 09

BETALING
21/11/2019 13:57 Auth. code: 019172
Totaal: 102,02 EUR
AKKOORD

2019

Bankrelatie:
IBAN: NL83 ABNA 04 9879 2757
IBAN: NL72 ABNA 05 1878 6110
AGB praktijk: 12001973
AGB declarant: 12064229
DECLARATIE: 97628/5811

BETREFT TANDHEELKUNDIGE BEHANDELING. Gaarne betaling binnen 4 weken onder vermelding van het declaratienumm

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
	21-11-2019	X10	27	1	Maken en beoordelen kleine röntgenfoto	16,05
	21-11-2019	V93	27	1	Drievlaksvulling composiet	71,64
	21-11-2019	A10	27	1	Geleidings-, infiltratie- en/of intraligamentaire verdoving	14,33
						Euro 102,02

Voldaan per pinbetaling

N. THEODOSIA
tandarts
Kastanjestraat 3
3203 AP Spijkenis
Tel. 0181

(b) Example of an image with high saturation level.

Figure 7.3: Examples of images with high levels of hue and saturation.

images as in Figure 7.4a (the mean equals to 0.34 and std to 0.6 with the average over all images of 1.3 and 22.4 respectively). However, low values of both features are occasionally obtained on sharp images, too. High values of mean are obtained on images of a screen as in the Figure 7.4b (the mean equals to 7, std — to 131). The standard deviation takes large values both on images of a screen and scans.

Then, S3 spatial and spectral sharpness measures from [102] are implemented. For spectral feature calculation, FFT is applied to an image and it is converted into polar coordinates image $\mathbf{x}$. Then, $z(f) = \sum_{\theta} \mathbf{x}(f, \theta)$ is computed, where θ is orientation and f is frequency. Afterwards, α^* given in eq. 7.9 is found numerically using Nelder-Mead algorithm provided by a Python package.

$$\alpha^* = \arg\min_{\alpha} ||f^{-\alpha} - z(f)||_2. \tag{7.9}$$

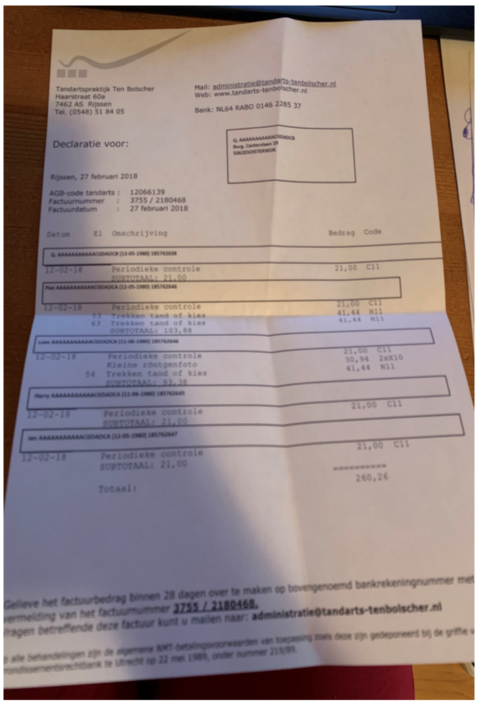
Finally, spectral sharpness measure $S_{spectral}$ is calculated as in eq. 7.10.

$$S_{spectral} = 1 - \frac{1}{1 + e^{-3(\alpha^* - 2)}}. \tag{7.10}$$

To calculate spatial S3 metric, the image is split into $20 \times 20 = 400$ regions $\mathbf{X}_j$. Then, total variance $v(\mathbf{X}_j)$ of each region is calculated according to eq. 7.11.

$$v(\mathbf{X}_j) = \sum_{i,j} |x_i - x_j|, \tag{7.11}$$

where i, j are all the pairs, so that x_i and x_j are neighbouring pixels.



Tandartspraktijk Ten Bolscher
Haarstraat 60a
7462 AS Rijssen
Tel. (0548) 51 84 05

Mail: administratie@tandarts-tenbolscher.nl
Web: www.tandarts-tenbolscher.nl

Bank: NL64 RABO 0146 2285 37

Declaratie voor:

Q. AAAAAAAAAACGDADCB
Burg. Canterslaan 29
5062ESOISTERWIJK

Rijssen, 27 februari 2018

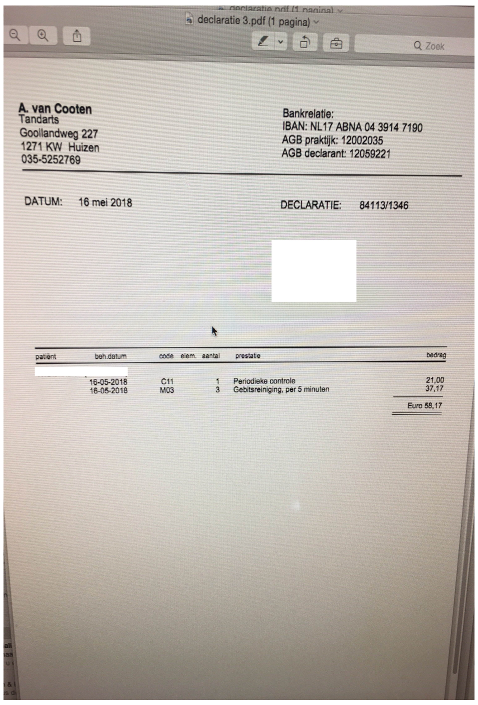
AGB-code tandarts	:	12066139
Factuurnummer	:	3755 / 2180468
Factuurdatum	:	27 februari 2018

Datum	El	Omschrijving	Bedrag	Code
Q. AAAAAAAAAAACGDADCB (13-05-1980) 185762638				
12-02-18		Periodieke controle	21,00	C11
		SUBTOTAAL: 21,00		
Piet AAAAAAAAAAACGDADCA (12-05-1980) 185762646				
12-02-18		Periodieke controle	21,00	C11
	53	Trekken tand of kies	41,44	H11
	63	Trekken tand of kies	41,44	H11
		SUBTOTAAL: 103,88		
Loes AAAAAAAAAAACGDADCA (11-06-1980) 185762646				
12-02-18		Periodieke controle	21,00	C11
		Kleine röntgenfoto	30,94	2xX10
	54	Trekken tand of kies	41,44	H11
		SUBTOTAAL: 93,38		
Harry AAAAAAAAAAACGDADCA (11-06-1980) 185762645				
12-02-18		Periodieke controle	21,00	C11
		SUBTOTAAL: 21,00		
Jan AAAAAAAAAAACGDADCA (12-05-1980) 185762647				
12-02-18		Periodieke controle	21,00	C11
		SUBTOTAAL: 21,00		
		Totaal:	260,26	

Gelieve het factuurbedrag binnen 28 dagen over te maken op bovengenoemd bankrekeningnummer met vermelding van het factuurnummer 3755 / 2180468.
Vragen betreffende deze factuur kunt u mailen naar: administratie@tandarts-tenbolscher.nl

Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing zoals deze zijn gedeponeerd bij de griffie van de arrondissementsrechtbank te Utrecht op 22 mei 1989, onder nummer 219/89.

(a) Example of an image with low mean and std of Laplace sharpness.



declaratie 3.pdf (1 pagina)
Zoek

A. van Cooten
Tandarts
Gooilandweg 227
1271 KW Huizen
035-5252769

Bankrelatie:
IBAN: NL17 ABNA 04 3914 7190
AGB praktijk: 12002035
AGB declarant: 12059221

DATUM: 16 mei 2018

DECLARATIE: 84113/1346

[redacted]

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
[redacted]						
	16-05-2018	C11		1	Periodieke controle	21,00
	16-05-2018	M03		3	Gebitsreiniging, per 5 minuten	37,17
						Euro 58,17

(b) Example of an image with high mean and std of Laplace sharpness.

Figure 7.4: Examples of images with high and low Laplace sharpness.

Afterwards, $S_{spatial} = \max_j v(\mathbf{X}_j)$ is taken as spatial S3 sharpness measure.

The spectral sharpness measure takes values in the interval $[0.999, 1]$ and does not provide a lot of information. This might be happening since we calculate spectral sharpness over the whole image and do not aggregate it over regions as suggested by the authors of [102]. If we calculate it over regions in the same manner as spatial sharpness, then it takes more than a minute to compute which is too long for the current project.

In contrast, the spatial S3 sharpness gives some good insights: low values are obtained for blurred images as in Figure 7.4a (13030 with the average feature value of 57927). High values are obtained for scans and screen images.

Furthermore, spatial S3 sharpness metric $S_{spatial}$ and both Laplace sharpness metrics are applied to 64×64 regions of an image and then, the standard deviation, the minimum, the maximum, and histograms over the whole image are used as features, too. We use 18 bins to cover the interval $(100000, 500000)$ and 2 more bins for values larger than 500000 and smaller than 100000 for $S_{spatial}$. Similarly, we define 30 bins for Laplace sharpness by covering the interval $(0, 3)$ for the mean and $(0, 30)$ for the standard deviation. In addition, we include the average of the mean values larger than 3 as a feature. All histograms are scaled so that they sum up to 1. This way, the sizes of an image do not influence them. Including these regional sharpness features will allow accounting for the difference in sharpness in different regions of the image.

One more sharpness metric implemented is FISH sharpness index from [103]. There, as discussed in Section 4.1.2, Daubechies 9/7 wavelet transform is applied to an image with 3 levels of decomposition resulting in S_{XY_n}, where $XY \in \{LH, HL, HH\}$ defines a subband and $n \in \{1, 2, 3\}$ — the level of decomposition. Then, log-energy for the subband XY_n is calculated as given in eq. 7.12.

$$E_{XY_n} = \log_{10}\left(1 + \frac{1}{N_n}\sum_{i,j} S^2_{XY_n}(i,j)\right), \tag{7.12}$$

where N_n is the number of the subband coefficients in the subbands on the nth level of decomposition.

The total log-energy for each level n is calculated as in eq. 7.13.

$$E_n = (1-\alpha)\frac{E_{LH_n} + E_{HL_n}}{2} + \alpha E_{HH_n}, \tag{7.13}$$

where $\alpha = 0.8$ is a parameter chosen empirically to give a larger weight to the HH subband.

Finally, the $FISH$ sharpness index is calculated as in eq. 7.14.

$$FISH = \sum_{n=1}^{3} 2^{3-n} E_n. \tag{7.14}$$

Low values of FISH index are obtained on blurred images such as in Fig. 7.4a with a value 3.7 (the mean FISH sharpness is 10). High values are obtained both on scans and screen images similarly to what was already observed for the Laplace sharpness metric.

Another implemented sharpness metric is s-index. It is calculated using the algorithm proposed in [59] but without the last step. The image is first periodized and dequantized using the algorithms from [71] and [27] respectively. Periodization is performed to cancel the border-to-border discontinuities. Dequantization is performed in order to avoid adding 0's to the total variance that correspond to the flat regions artificially created by quantization of the image to $\{0, 1, \ldots, 255\}$ and add small numbers instead.

The periodized image $per(u)$ of the original image u is obtained as given in eq. 7.15.

$$per(u) = u - s, \tag{7.15}$$

where $\hat{s}$ (a FFT of s) is given by eq. 7.16.

$$\hat{s}(q,r) = \frac{\hat{v}(q,r)}{4 - 2\cos\frac{2\pi r}{N} - 2\cos\frac{2\pi q}{M}}, \tag{7.16}$$

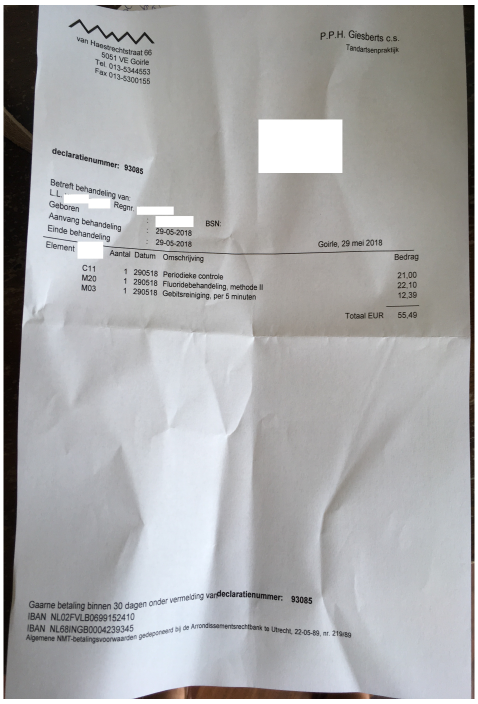
where M, N are the sizes of the image and $v = u$ on all the pixels except the borders where v equals to the difference between the opposite border pixels. $\hat{v}$ is a FFT of v.

Dequantized image $\tau(u)$ is defined as inverse FFT of $\widehat{\tau(u)}$ given in eq. 7.17.

$$\widehat{\tau(u)}(q,r) = \widehat{per(u)}(q,r)e^{-i\pi\left(\frac{q}{M}+\frac{r}{N}\right)}, \tag{7.17}$$

where $\widehat{per(u)}$ is a FFT of $per(u)$.

The further algorithm can be generally described as following:



van Haestrechtstraat 66
5051 VE Goirle
Tel. 013-5344553
Fax 013-5300155

P.P.H. Giesberts c.s.
Tandartsenpraktijk

[redacted]

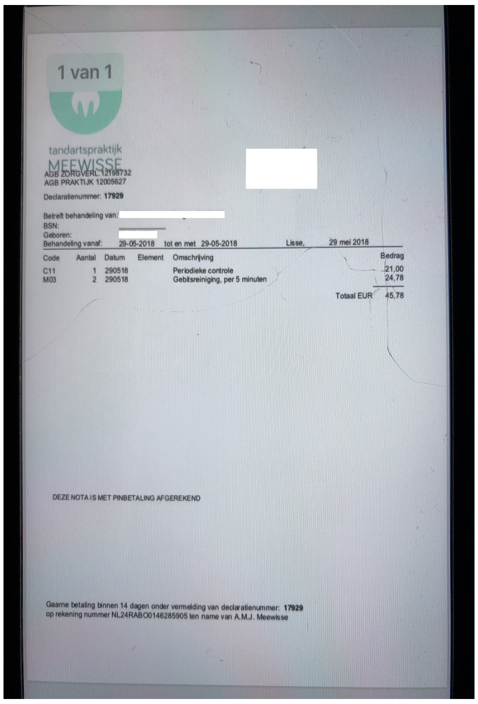
declaratienummer: 93085

Betreft behandeling van:
L.L. [redacted] [redacted] Regnr. [redacted]
Geboren : [redacted] BSN:
Aanvang behandeling : 29-05-2018
Einde behandeling : 29-05-2018 Goirle, 29 mei 2018

Element [redacted]	Aantal	Datum	Omschrijving	Bedrag
C11	1	290518	Periodieke controle	21,00
M20	1	290518	Fluoridebehandeling, methode II	22,10
M03	1	290518	Gebitsreiniging, per 5 minuten	12,39
			Totaal EUR	55,49

Gaarne betaling binnen 30 dagen onder vermelding van **declaratienummer: 93085**
IBAN NL02FVLB0699152410
IBAN NL68INGB0004239345
Algemene NMT-betalingsvoorwaarden gedeponeerd bij de Arrondissementsrechtbank te Utrecht, 22-05-89, nr. 219/89

(a) Example of an image with a high s-index.



tandartspraktijk
MEEWISSE
AGB ZORGVERL. 12198732
AGB PRAKTIJK 12005627

Declaratienummer: **17929**

[redacted]

Betreft behandeling van: [redacted]
BSN:
Geboren: [redacted]
Behandeling vanaf: 29-05-2018 tot en met 29-05-2018 Lisse, 29 mei 2018

Code	Aantal	Datum	Element	Omschrijving	Bedrag
C11	1	290518		Periodieke controle	21,00
M03	2	290518		Gebitsreiniging, per 5 minuten	24,78
				Totaal EUR	45,78

DEZE NOTA IS MET PINBETALING AFGEREKEND

Gaarne betaling binnen 14 dagen onder vermelding van declaratienummer: **17929**
op rekening nummer NL24RABO0146285905 ten name van A.M.J. Meewisse

(b) Example of an image with a low s-index.

Figure 7.5: Examples of images with high and low s-index.

1. Calculate total variance TV of the whole image $\tau(u)$ similarly to eq. 7.11;
2. Calculate vertical and horizontal gradient images u_y and u_x of the converted image $\tau(u)$ using derivative filters;
3. Apply FFT to both gradient images to obtain $\hat{u}_x$ and $\hat{u}_y$;
4. Construct new images Γ_i with $i \in \{xx, xy, yy\}$ as an element-wise product of the absolute values of the respective transformed gradient images;
5. Calculate μ, σ^2 according to eq. 7.18-7.19;
6. Calculate a sharpness index S_{ind} as in eq. 7.20.

$$\mu = (\alpha_x + \alpha_y)\sqrt{\frac{2MN}{\pi}}. \tag{7.18}$$

$$\sigma^2 = \frac{1}{\pi MN}\left(\frac{||\Gamma_{xx}||_2^2}{\alpha_x^2} + 2\frac{||\Gamma_{xy}||_2^2}{\alpha_x \alpha_y} + \frac{||\Gamma_{yy}||_2^2}{\alpha_y^2}\right), \tag{7.19}$$

where α_x and α_y — 2-norms of the horizontal and vertical gradient images respectively and M, N — sizes of the image.

$$S_{ind} = \frac{\mu - TV}{\sigma}. \tag{7.20}$$

S-index can be as well a useful sharpness feature for the classification as the minimum values of s-index are obtained on images of a screen as in Fig. 7.5b with a value of 7 (the minimum value) and high values are obtained on visually sharp images which are not scans as in Fig. 7.5a with a value of 724 (the maximum value, the mean value of s-index is 303).

Thus, in this section, we have discussed in detail the implementations of extraction of the features that correspond to various image characteristics such as brightness, contrast, and sharpness. We have looked into the examples of the images with high and low values of these features and discussed how these features can help in the classification.

7.2 General image quality measures

After extraction of the features which represent image characteristics, we move to the extraction of the features which are used in state-of-the-art IQA algorithms. As mentioned in Section 4.1.2, we can use features from general IQA algorithms without the last step, which is performed to estimate overall quality. This makes sense as we create our own analogue of IQA algorithm and, thus, want to use features rather than single quality estimates. Besides this, we can't apply this last step for opinion-aware metrics as we do not have subjectively evaluated data.

We implement the features from BRISQUE ([70]) and IL-NIQE ([114]) algorithms. In BRISUQE, there are 18 features used which are parameters of 4 asymmetric GGD (AGGD) and one GGD. Firstly, MSCN (mean subtracted contrast normalized) coefficients $\hat{I}(i,j)$ are calculated according to eq. 7.21.

$$\hat{I}(i,j) = \frac{I(i,j) - \mu(i,j)}{\sigma(i,j) + 1}, \tag{7.21}$$

where I is the gray-scale image, local mean field μ is the image I blurred with a Gaussian filter of size 3×3 and the local variance field σ^2 is the image $(I - \mu)^2$ also blurred with a Gaussian filter.

Then, products of MSCN coefficients along 4 different orientations (horizontal, vertical and 2 diagonals) are calculated as in eq. 7.22-7.25.

$$H(i,j) = \hat{I}(i,j)\hat{I}(i,j+1), \tag{7.22}$$

$$V(i,j) = \hat{I}(i,j)\hat{I}(i+1,j), \tag{7.23}$$

$$D1(i,j) = \hat{I}(i,j)\hat{I}(i+1,j+1), \tag{7.24}$$

$$D2(i,j) = \hat{I}(i,j)\hat{I}(i+1,j-1). \tag{7.25}$$

MSCN coefficients are assumed to obey to GGD while H, V, $D1$, and $D2$ are assumed to obey to AGGD with probability density function (PDF) given in eq. 7.26. PDF of GGD has the same form with a difference that $\sigma_l = \sigma_r$ and, hence, its PDF has a from of eq. 7.26 with only one case.

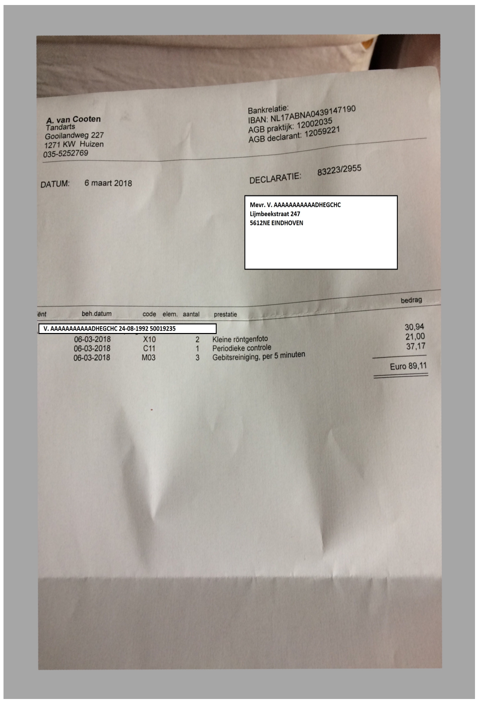
$$f(x;\nu,\sigma_l,\sigma_r) = \begin{cases} \frac{\nu}{(\beta_l+\beta_r)\Gamma\left(\frac{1}{\nu}\right)} e^{-\left(\frac{-x}{\beta_l}\right)^{\nu}} & x < 0, \\ \frac{\nu}{(\beta_l+\beta_r)\Gamma\left(\frac{1}{\nu}\right)} e^{-\left(\frac{x}{\beta_r}\right)^{\nu}} & x \geq 0, \end{cases} \tag{7.26}$$

where

$$\beta_l = \sigma_l \frac{\sqrt{\Gamma\left(\frac{1}{\nu}\right)}}{\sqrt{\Gamma\left(\frac{3}{\nu}\right)}}, \tag{7.27}$$

$$\beta_r = \sigma_r \frac{\sqrt{\Gamma\left(\frac{1}{\nu}\right)}}{\sqrt{\Gamma\left(\frac{3}{\nu}\right)}}. \tag{7.28}$$

Then, flattened MSCN coefficients are fitted to GGD and parameters ν and σ are approximated using the moment-based matching approach from [57]. Alongside with it, H, V, $D1$ and $D2$ are fitted to AGGD and the parameters ν, σ_l, σ_r and its mean are approximated by the same method.


A. van Cooten
Tandarts
Gooilandweg 227
1271 KW Huizen
035-5252769

Bankrelatie:
IBAN: NL17ABNA0439147190
AGB praktijk: 12002035
AGB declarant: 12059221

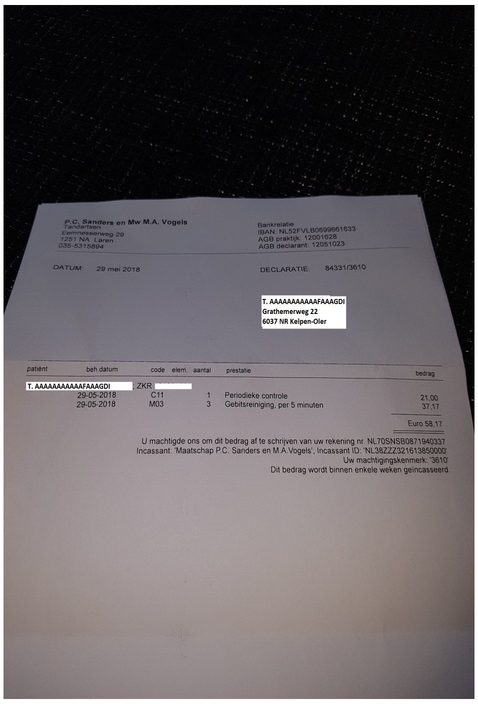
DATUM: 6 maart 2018

DECLARATIE: 83223/2955

Mevr. V. AAAAAAAAAAADHEGCHC
Lijmbeekstraat 247
5612NE EINDHOVEN

ënt	beh.datum	code	elem.	aantal	prestatie	bedrag
V. AAAAAAAAAAADHEGCHC 24-08-1992 50019235						
	06-03-2018	X10		2	Kleine röntgenfoto	30,94
	06-03-2018	C11		1	Periodieke controle	21,00
	06-03-2018	M03		3	Gebitsreiniging, per 5 minuten	37,17
						Euro 89,11

(a) Example of an image with a low BRISQUE σ value for MSCN coefficients.


P.C. Sanders en Mw M.A. Vogels
Tandartsen
Eemnesserweg 29
1251 NA Laren
035-5315894

Bankrelatie
IBAN: NL52FVLB0699661633
AGB praktijk: 12001628
AGB declarant: 12051023

DATUM: 29 mei 2018

DECLARATIE: 84331/3610

T. AAAAAAAAAAAFAAAGDI
Grathemerweg 22
6037 NR Kelpen-Oler

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
T. AAAAAAAAAAAFAAAGDI ; ZKR [redacted]						
	29-05-2018	C11		1	Periodieke controle	21,00
	29-05-2018	M03		3	Gebitsreiniging, per 5 minuten	37,17
						Euro 58,17

U machtigde ons om dit bedrag af te schrijven van uw rekening nr. NL70SNSB0871940337
Incassant: 'Maatschap P.C. Sanders en M.A.Vogels', Incassant ID: 'NL38ZZZ321613850000'
Uw machtigingskenmerk: '3610'
Dit bedrag wordt binnen enkele weken geïncasseerd.

(b) Example of an image with a high BRISQUE σ value for MSCN coefficients.

Figure 7.6: Examples of images for high and low values of BRISQUE feature σ for MSCN coefficients.

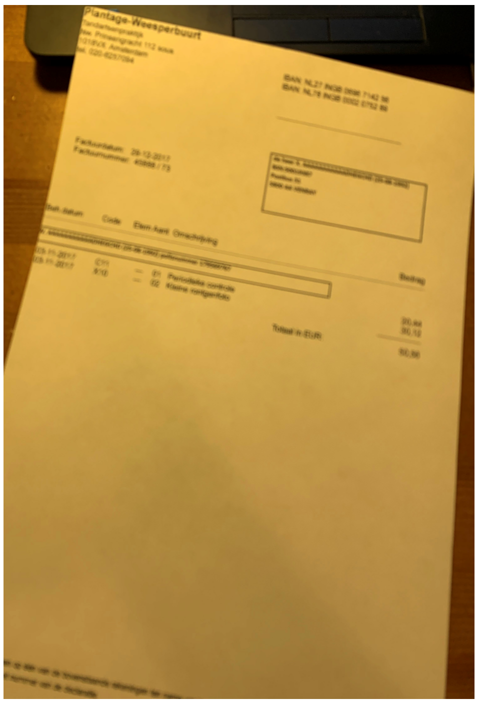
BRISQUE shape features ν distinguish well scans and images of a screen. For example, one of the smallest values of shape parameter $\nu = 0.3$ for MSCN coefficients is obtained on the scan in Figure 7.1a and one of the highest $\nu = 25.8$ is obtained on Figure 7.4b, which appeared also for the Laplace features. Other BRISQUE features do not have that obvious explanation.

For instance, the images presented in Figure 7.6 have a low value $\sigma = 0.28$ (Fig. 7.6a) and a high value $\sigma = 0.72$ (Fig 7.6b) of σ parameter of distribution of MSCN coefficients.

Let's move to the next investigated IQA method which is IL-NIQE. IL-NIQE method uses BRISQUE features and, besides them, introduces new ones. To extract some of these features, image transformations are required. Firstly, the original image I is converted from RGB channels into $O_1O_2O_3$ channels by the transformation in eq. 7.29 resulting in the image $\hat{I}_O$.

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} 0.06 & 0.63 & 0.27 \\ 0.3 & 0.04 & -0.35 \\ 0.34 & -0.6 & 0.17 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \tag{7.29}$$

Then, vertical and horizontal derivative filters or Sobel filters are applied to $\hat{I}_O$ to obtain gradients of the image, $\hat{I}_{Ov}$ and $\hat{I}_{Oh}$. The difference between the filters is that the Sobel filter provides the smooth gradient of the image while the derivative filter does not. However, it was revealed that the type of filter does not influence the classification algorithm performance much. Thus, it was decided to apply derivative filters.



Plantage-Weesperbuurt
Tandartsenpraktijk
Nw. Prinsengracht 112 [illegible]
1018VX Amsterdam
tel. 020-6257094

IBAN NL27 INGB [illegible]
IBAN NL76 INGB [illegible]

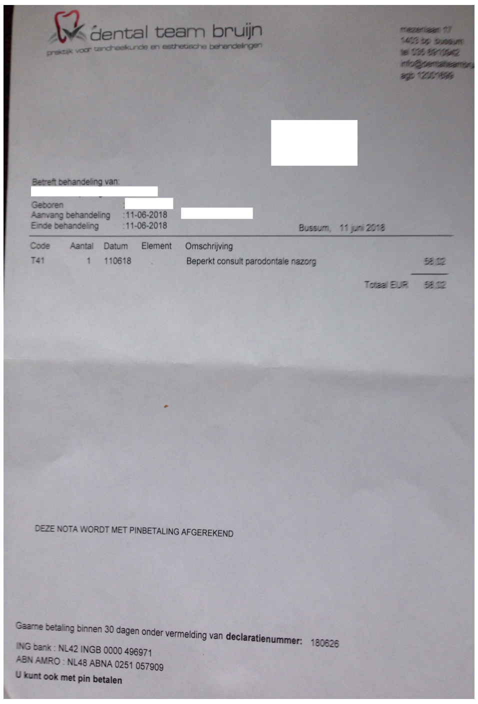
Factuurdatum: 29-12-2017
Factuurnummer: [illegible]

[illegible]

Beh.datum	Code	Elem Aant	Omschrijving	Bedrag
03-11-2017	C11	— 01	Periodieke controle	[illegible]
03-11-2017	X10	— 02	Kleine röntgenfoto	[illegible]

Totaal in EUR [illegible]

(a) The image with the minimum value of IL-NIQE horizontal gradient feature $\sigma = 535$ for the channel O_1 (the mean is 8657).



dental team bruijn
praktijk voor tandheelkunde en esthetische behandelingen

[illegible] 17
1403 [illegible] bussum
tel 035 [illegible]
info@[illegible]
agb [illegible]

[redacted]

Betreft behandeling van:
[redacted]
Geboren : [redacted]
Aanvang behandeling :11-06-2018 [redacted]
Einde behandeling :11-06-2018

Bussum, 11 juni 2018

Code	Aantal	Datum	Element	Omschrijving	
T41	1	110618	.	Beperkt consult parodontale nazorg	58,02
				Totaal EUR	58,02

DEZE NOTA WORDT MET PINBETALING AFGEREKEND

Gaarne betaling binnen 30 dagen onder vermelding van **declaratienummer:** 180626
ING bank : NL42 INGB 0000 496971
ABN AMRO : NL48 ABNA 0251 057909
U kunt ook met pin betalen

(b) The image with the maximum value of IL-NIQE horizontal gradient feature $\nu = 0.51$ for the channel O_2 (the mean is 0.27).

Figure 7.7: The images with the maximum and minimum values of some IL-NIQE horizontal gradient features.

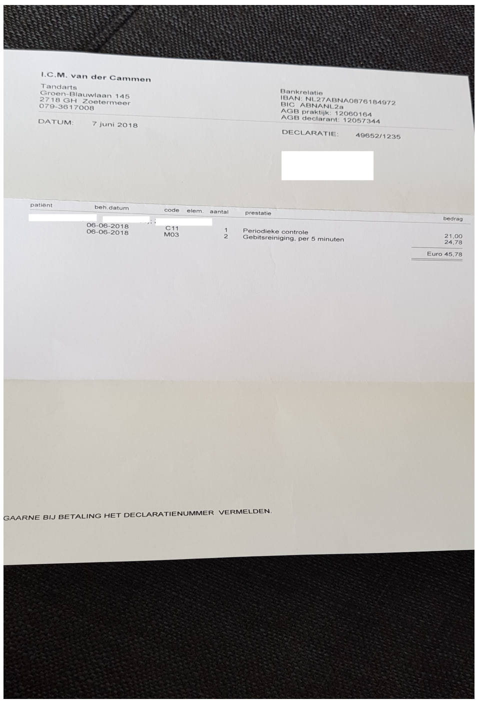
Afterwards, for each of the channels $O_1O_2O_3$ of $\hat{I}_{Ov}$ and $\hat{I}_{Oh}$ MSCN coefficients are calculated and parameters ν and σ of fitted GGD are approximated resulting in 12 features. We will refer to them as to IL-NIQE horizontal and vertical gradient features. These features perform well in evaluating sharpness. For example, the maximum values of the shape parameter ν

for all channels O_1, O_2, and O_3 are obtained on screen images and blurred images and the minimum values are obtained on scans. The variance parameters σ distinguish blurred images well, too. Their minimum values are obtained on blurred images and the maximum values are obtained on scans. The examples of blurred images that can be detected using these features are given in Figure 7.7. Thus, we can see that the IL-NIQE horizontal gradient features are helpful when evaluating blur in images.

Next, 6 IL-NIQE color features are extracted using RGB color channels. These features are the means and standard deviations of l_1, l_2 and l_3 defined as in eq. 7.30-7.32.

$$l_1 = (\log R - \mu_R + \log G - \mu_G + \log B - \mu_B)/\sqrt{3}, \tag{7.30}$$
$$l_2 = (\log R - \mu_R + \log G - \mu_G - 2\log B + 2\mu_B)/\sqrt{6}, \tag{7.31}$$
$$l_3 = (\log R - \mu_R - \log G + \mu_G)/\sqrt{2}, \tag{7.32}$$

where μ_R, μ_G and μ_B represent the means of $\log R$, $\log G$ and $\log B$ respectively.



I.C.M. van der Cammen
Tandarts
Groen-Blauwlaan 145
2718 GH Zoetermeer
079-3617008

Bankrelatie
IBAN: NL27ABNA0876184972
BIC ABNANL2a
AGB praktijk: 12060164
AGB declarant: 12057344

DATUM: 7 juni 2018

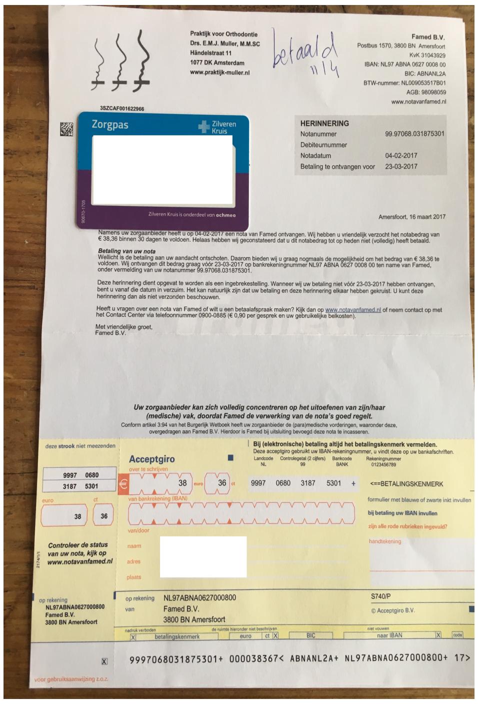
DECLARATIE: 49652/1235

[redacted]

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
[redacted]	[redacted]	[redacted]				
	06-06-2018	C11		1	Periodieke controle	21,00
	06-06-2018	M03		2	Gebitsreiniging, per 5 minuten	24,78
						Euro 45,78

GAARNE BIJ BETALING HET DECLARATIENUMMER VERMELDEN.

(a) Example of an image with a low value of IL-NIQE mean color feature for l_1.



Praktijk voor Orthodontie
Drs. E.M.J. Muller, M.M.SC
Händelstraat 11
1077 DK Amsterdam
www.praktijk-muller.nl

betaald
11/4

Famed B.V.
Postbus 1570, 3800 BN Amersfoort
KvK 31043929
IBAN: NL97 ABNA 0627 0008 00
BIC: ABNANL2A
BTW-nummer: NL009053517B01
AGB: 98098059
www.notavanfamed.nl

3SZCAF001622966

Zorgpas
Zilveren Kruis
[redacted]
Zilveren Kruis is onderdeel van achmea
90670-1705

HERINNERING	
Notanummer	99.97068.031875301
Debiteurnummer	
Notadatum	04-02-2017
Betaling te ontvangen voor	23-03-2017

Amersfoort, 16 maart 2017

Namens uw zorgaanbieder heeft u op 04-02-2017 een nota van Famed ontvangen. Wij hebben u vriendelijk verzocht het notabedrag van € 38,36 binnen 30 dagen te voldoen. Helaas hebben wij geconstateerd dat u dit notabedrag tot op heden niet (volledig) heeft betaald.

Betaling van uw nota
Wellicht is de betaling aan uw aandacht ontschoten. Daarom bieden wij u graag nogmaals de mogelijkheid om het bedrag van € 38,36 te voldoen. Wij ontvangen dit bedrag graag vóór 23-03-2017 op bankrekeningnummer NL97 ABNA 0627 0008 00 ten name van Famed, onder vermelding van uw notanummer 99.97068.031875301.

Deze herinnering dient opgevat te worden als een ingebrekestelling. Wanneer wij uw betaling niet vóór 23-03-2017 hebben ontvangen, bent u vanaf die datum in verzuim. Het kan natuurlijk zijn dat uw betaling en deze herinnering elkaar hebben gekruist. U kunt deze herinnering dan als niet verzonden beschouwen.

Heeft u vragen over een nota van Famed of wilt u een betaalafspraak maken? Kijk dan op www.notavanfamed.nl of neem contact op met het Contact Center via telefoonnummer 0900-0885 (€ 0,90 per gesprek en uw gebruikelijke belkosten).

Met vriendelijke groet,
Famed B.V.

Uw zorgaanbieder kan zich volledig concentreren op het uitoefenen van zijn/haar (medische) vak, doordat Famed de verwerking van de nota's goed regelt.
Conform artikel 3:94 van het Burgerlijk Wetboek heeft uw zorgaanbieder de (para)medische vorderingen, waaronder deze, overgedragen aan Famed B.V. Hierdoor is Famed bij uitsluiting bevoegd deze nota te incasseren.

deze **strook** niet meezenden
9997 0680
3187 5301
euro 38 ct 36

Controleer de status van uw nota, kijk op www.notavanfamed.nl
2174/1/1

op rekening
NL97ABNA0627000800
Famed B.V.
3800 BN Amersfoort

Acceptgiro
Bij (elektronische) betaling altijd het betalingskenmerk vermelden.
Deze acceptgiro gebruikt uw IBAN-rekeningnummer, u vindt deze op uw bankafschriften.

Landcode	Controlegetal (2 cijfers)	Bankcode	Rekeningnummer
NL	99	BANK	0123456789

over te schrijven
€ 38 euro 36 ct
9997 0680 3187 5301 + <==BETALINGSKENMERK
van bankrekening (IBAN)
formulier met blauwe of zwarte inkt invullen
bij betaling uw IBAN invullen
zijn alle rode rubrieken ingevuld?
van/door
naam [redacted]
adres
plaats
handtekening

op rekening NL97ABNA0627000800
van Famed B.V.
3800 BN Amersfoort
S740/P
© Acceptgiro B.V.

nadruk verboden | de ruimte hieronder niet beschrijven | niet vouwen
X betalingskenmerk | euro | ct X | BIC | naar IBAN X | code

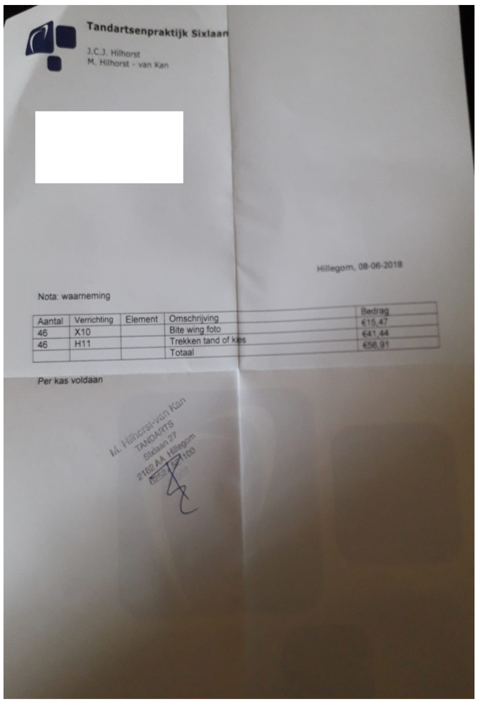
X 9997068031875301+ 000038367< ABNANL2A+ NL97ABNA0627000800+ 17>

voor gebruiksaanwijzing z.o.z.

(b) Example of an image with a high value of IL-NIQE mean color feature for l_1.

Figure 7.8: Examples of images for high and low values of IL-NIQE color features.

Low values of the mean color features are obtained on scans and images with a prevalence of gray color as in Figure 7.8a. The mean color feature for l_1 in this image is -0.627 while the average value of this feature is 0.281. High values are obtained on images with a wide range of colors or images with a not gray but some other prevalent color: for instance, pink or blue. An example of a large color mean equal to 0.87 can be found in Figure 7.8b. The standard deviation features also take the minimum values on scans and images with a prevalence of one color. Large values are obtained if the image has, for example, a black or blue background which contrasts with the white color of the paper. Thus, IL-NIQE color features distinguish several interesting color properties of images. However, the colors do not play a significant role in OCR algorithms, so, these features might be not very useful for the classification.



Tandartsenpraktijk Sixlaan

J.C.J. Hilhorst
M. Hilhorst - van Kan

[redacted]

Hillegom, 08-06-2018

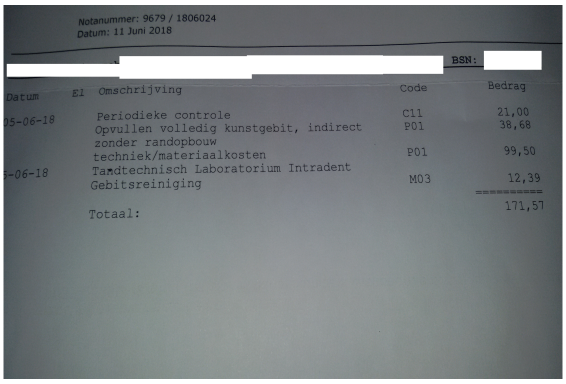
Nota: waarneming

Aantal	Verrichting	Element	Omschrijving	Bedrag
46	X10		Bite wing foto	€15,47
46	H11		Trekken tand of kies	€41,44
			Totaal	€56,91

Per kas voldaan

M. Hilhorst-van Kan
TANDARTS
Sixlaan 27
2182 AA Hillegom
0252-521100

(a) Example of an image with a low value of the feature.



Notanummer: 9679 / 1806024
Datum: 11 Juni 2018

[redacted] BSN: [redacted]

Datum	El	Omschrijving	Code	Bedrag
05-06-18		Periodieke controle	C11	21,00
		Opvullen volledig kunstgebit, indirect zonder randopbouw	P01	38,68
		techniek/materiaalkosten	P01	99,50
5-06-18		Tandtechnisch Laboratorium Intradent		
		Gebitsreiniging	M03	12,39
		Totaal:		171,57

(b) Example of an image with a high value of the feature.

Figure 7.9: Examples of images with low and high values of simple IL-NIQE Gabor feature ν for the orientation 0 and scale 1/2.

Finally, 150 features are calculated by applying log-Gabor filters. We apply FFT to the grayscaled version of the original image and transform it into polar coordinates. Then, it is multiplied by a set of log-Gabor kernels (eq. 4.3) of the same size as the image with $\sigma_r = 0.996\sqrt{2/3}$, $\sigma_\theta = 0.996\pi/5\sqrt{2}$, and all possible combinations of 5 different central orientations (0, $\pi/5$, $2\pi/5$, $3\pi/5$, $4\pi/5$) and 5 different scales ($1/2^n$, where $n \in \{1, \dots, 5\}$. Here, high values of the central scale correspond to high-frequency structures such as edges, and low values — to low-frequency structures such as blurred edges or noise. The central orientation represents the direction in which the sharpness of the image is assessed. Furthermore, vertical and horizontal gradient images are constructed from the grayscaled original image using respective derivative filters. Then, they are also transformed into polar coordinates and multiplied by the same set of log-Gabor kernels. After application of log-Gabor filters, the images are transformed back to Cartesian coordinates and back from frequency domain using inverse DFT. Next, the obtained real part of transformed images is flattened and parameters ν and σ of fitted GGD are approximated. This results in $3 \times (5 \times 5)$ sets of parameters and, hence, 150 features. 50 of them which were obtained from the grayscaled original image we will call simple IL-NIQE Gabor features. Other 100 we will call horizontal and vertical gradient IL-NIQE Gabor features respectively. In the original paper [114], the authors also suggested fitting GGD to the imaginary part of the transformed images. However, we observed a very high correlation between approximated parameters of GGD for the real and imaginary parts, thus, we decided to keep only the ones related to the real part.

We can notice that some of the IL-NIQE Gabor features can capture the difference in sharpness. For example, we obtain one of the 5 smallest values (0.25 with mean of this feature of 0.55) for the simple IL-NIQE Gabor ν feature with the orientation 0 and scale 1/2 for the image in Figure 7.9a. However, high values of this feature are obtained on various kinds of images: screen images, scans, and sharp images as in Figure 7.9b (equals to 0.885) which does not provide useful information for the classification. Another example of a blurred image for which we obtain extreme values of some IL-NIQE Gabor feature is given in Figure 7.7a. There, we obtain the minimum value of the simple IL-NIQE Gabor σ feature with the orientation 0 and scale 1/8 (equals 0.45, the mean of this feature is 31.5). The maximum values of this feature are obtained on scans. Therefore, it also succeeds in distinguishing blurred images and scans.

Thus, in this section, we have discussed the features which are used in BRISQUE and IL-NIQE IQA method and elaborated on their implementations. Furthermore, we have shown that these features can differentiate between various kinds of images such as, for instance: scans and screen photos, sharp and blurred images, images with a wide range of colors and images with a prevalence of a single color. Therefore, these features might add value to the classification algorithm.

7.3 Document-specific image quality features

In the section above, we have discussed features that can be relevant for any kind of images. In this section, we will discuss the document-specific features.

7.3.1 Skew evaluation

One of the document-specific features is text skew. To evaluate it, several methods from Section 4.3.1 are implemented.

Before applying the methods, an image is downscaled by factor 4, binarized, rotated to correct orientation, and cropped by 10% from each side. Downscaling is done to reduce the computational cost, and binarization is done as chosen skew detection methods use a binary image as an input. Cropping is performed in order to minimize the presence of paper edges on the image and try to retain only text.

Then, 7 methods of skew detection are applied. Some of these methods are applied not only to the whole transformed image but also to the 32×32 regions of it in order to account for the possibility of different orientations in different regions of the image. Only the regions with a sufficient amount of black pixels (more than $3 - 5\%$ of the region) are considered for regional skew evaluation. Then, the mean, the standard deviation, and the maximum absolute value of the regional values are taken as features. This results in 3 features which we will refer to as general regional skew features and will extract for all skew detection methods. Besides them, the histograms of regional values are constructed and also taken as features which gives from 9 to 15 features — depending on the number of bins. The bins were chosen as $[< -b, -b, -b+1, \ldots, 0, 1, 2, \ldots, b, > b]$ for $b \in \{3, 6\}$. This split into bins provided the highest number of important features compared to other investigated splits.

The first implemented method is the one using the Hough transform from [43] which is discussed in detail in Section 4.3.1. It is implemented both for bursted and non-bursted images (resulting in 2 features) as well as for regions 32×32 of the non-bursted image (resulting in 3 general regional skew features and 9 histogram features for split into the bins with $b = 3$). Its outcome is not trustworthy as for some examples of scanned invoices it returns various large skew angles instead of values close to 0 (Tab. 7.1). This is expected as this method requires dense text on the image. However, we keep these 14 features as they still might provide some useful information.

Two more implemented methods: with use of the DFT as in [85] and with the use of connected components as in [79] do not provide with reliable results and, hence, were not used further.

Next implemented methods provided more reliable estimates. They are the complexity variance (CV) method from [55], the projection profile (PP) method from [23], the morphological method from [25], and the Sauvola method using Rao's formula from [94].

The complexity variance and projection profile methods are applied to the whole image resulting in 1 feature each, and to the regions of it resulting in 3 general regional features for each method, 9 histogram features for CV with $b = 3$, and 15 histogram features for PP with $b = 6$. These methods were already explained in Section 4.3.1.

In the morphological method, we first apply a morphological closing operation with a line structuring element 18×1 and then, an opening with a square structure element 5×5. Let's define these morphological operations. The closing operation is a dilation followed by an erosion. The opening operation is an erosion followed by a dilation. By definition, the erosion transforms each pixel of the image into 1 if the structuring element centered in this pixel fits the input image, meaning that all the covered image pixels are equal to 1. The dilation is a similar operation with the difference that the structuring element has to hit the input image, meaning that at least one of the covered image pixels should be equal to 1. In our project, pixels that equal to 1 are black pixels. There exists a Python library that provides morphological operations on images, so there is no need to implement them manually.

After applying the morphological operations, a so-called transfer image is constructed by keeping only black pixels followed by white in the vertical direction. Afterwards, lines consisting of neighbouring pixels are identified. Finally, the skew angle of each line of sufficient length (more than 8 pixels) is calculated. In the same manner as with regions, we calculate the mean, the standard deviation, and the maximum of the absolute value of these skew angles as features (resulting in 3 features) and constructing histogram with $b = 3$, which gives 9 more features.

One more used method, a so-called Sauvola method by the author name, is introduced in [94]. In this method, the image is filtered in vertical and horizontal directions with line filters $h_x(i, j)$ and $h_y(i, j)$ given in eq. 7.33-7.34 resulting in $\hat{I}_x$ and $\hat{I}_y$ respectively.

$$h_x(i,j) = \frac{2i}{\sigma^2} e^{-\frac{(i^2+j^2)}{\sigma^2}}, \tag{7.33}$$

$$h_y(i,j) = \frac{2j}{\sigma^2} e^{-\frac{(i^2+j^2)}{\sigma^2}}, \tag{7.34}$$

where $-s < i, j < s$ with $s \approx \left[\sigma\sqrt{-2\log\sigma - \log(0.005)}\right]$.

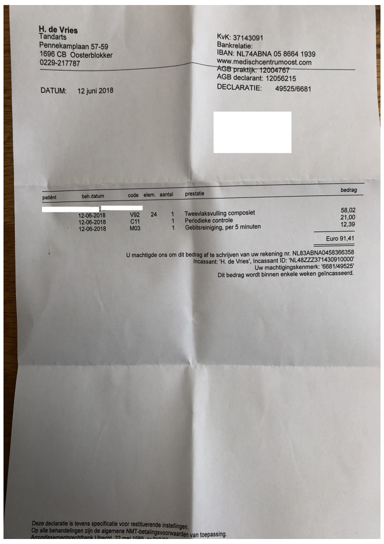
Afterwards, a gradient vector (G, ϕ) is calculated with $G = \sqrt{\hat{I}_x^2 + \hat{I}_y^2}$ and $\phi = \arctan(\hat{I}_x/\hat{I}_y)$. Finally, a Rao's formula (eq. 7.35) is applied to the windows of size $m \times m$ with $m = 10$ in order to find skew angles θ for each window.

$$\theta_w = \frac{1}{2}\left(\arctan\frac{\sum_{i=1}^{m}\sum_{j=1}^{m} G(i,j)^2\sin(2\phi(i,j))}{\sum_{i=1}^{m}\sum_{j=1}^{m} G(i,j)^2\cos(2\phi(i,j))}\right). \tag{7.35}$$

Then, as before, we use the mean, the standard deviation, and the maximum absolute value of all θ_w (resulting in 3 features) and histogram features with $b = 3$ (resulting in 9 features). In addition, we include the most frequent θ value as a feature as this is how the authors define the skew angle of a document in [94].

Method \ Image	Fig. 7.10a		Fig. 7.10b		Fig. 7.10c	
	angle	std	angle	std	angle	std
Hough	-15	11.9	-13	12.9	4	13.3
CV	0	8.4	38	7.2	0	12.6
Morphology	-1	4.8	-3	3.9	-6	8.9
PP	-1	2.6	-5	3.4	-5	9.6
Sauvola	-2	12.8	-3	10	0	10.4

Table 7.1: The skew angles for some examples (positive skew angle corresponds to counter-clockwise rotation).


H. de Vries
Tandarts
Pennekamplaan 57-59
1696 CB Oosterblokker
0229-217787

KvK: 37143091
Bankrelatie:
IBAN: NL74ABNA 05 8664 1939
www.medischcentrumoost.com
AGB praktijk: 12004767
AGB declarant: 12056215
DECLARATIE: 49525/6681

DATUM: 12 juni 2018

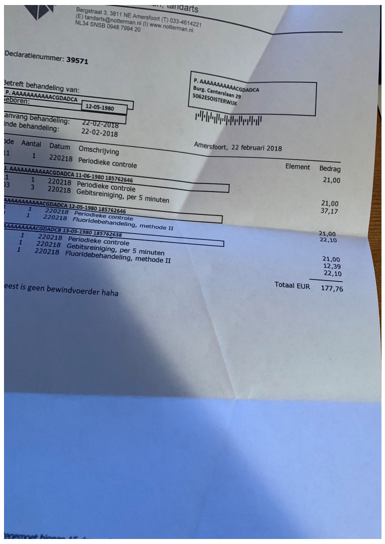
[redacted]

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
[redacted]	12-06-2018	V92	24	1	Tweevlaksvulling composiet	58,02
	12-06-2018	C11		1	Periodieke controle	21,00
	12-06-2018	M03		1	Gebitsreiniging, per 5 minuten	12,39
						Euro 91,41

U machtigde ons om dit bedrag af te schrijven van uw rekening nr. NL83ABNA0458366358
Incassant: 'H. de Vries', Incassant ID: 'NL48ZZZ371430910000'
Uw machtigingskenmerk: '6681/49525'
Dit bedrag wordt binnen enkele weken geïncasseerd.

Deze declaratie is tevens specificatie voor restituerende instellingen.
Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing.
Arrondissementsrechtbank Utrecht, 22 mei 1989

(a) Example of an image with no skew.


, tandarts
Bergstraat 3, 3811 NE Amersfoort (T) 033-4614221
(E) tandarts@notterman.nl (I) www.notterman.nl
NL34 SNSB 0948 7994 20

Declaratienummer: **39571**

P. AAAAAAAAAAACGDADCA
Burg. Canterslaan 29
5062ESOISTERWIJK

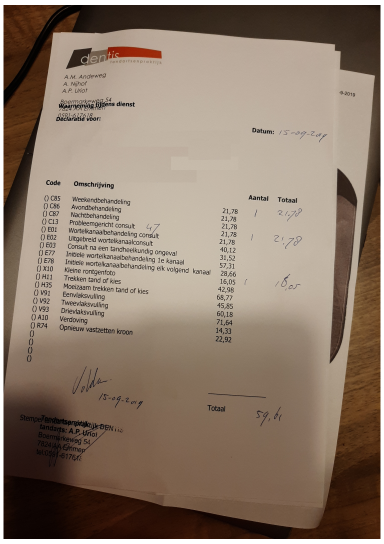
Betreft behandeling van:
P. AAAAAAAAAAACGDADCA
eboren: 12-05-1980
anvang behandeling: 22-02-2018
inde behandeling: 22-02-2018

Amersfoort, 22 februari 2018

de	Aantal	Datum	Omschrijving	Element	Bedrag
11	1	220218	Periodieke controle		21,00
J. AAAAAAAAAAACGDADCA 11-06-1980 185762646					
1	1	220218	Periodieke controle		21,00
03	3	220218	Gebitsreiniging, per 5 minuten		37,17
AAAAAAAAAAACGDADCA 12-05-1980 185762646					
	1	220218	Periodieke controle		21,00
	1	220218	Fluoridebehandeling, methode II		22,10
AAAAAAAAAACGDADCB 13-05-1980 185762638					
	1	220218	Periodieke controle		21,00
	1	220218	Gebitsreiniging, per 5 minuten		12,39
	1	220218	Fluoridebehandeling, methode II		22,10
				Totaal EUR	177,76

eest is geen bewindvoerder haha

(b) Example of an image with skew present.


dentis tandartsenpraktijk
A.M. Andeweg
A. Nijhof
A.P. Uriot
Boermarkeweg 54
7824 AA Emmen
0591-617618
Waarneming tijdens dienst
Declaratie voor:

Datum: 15-09-2019

[redacted]

-9-2019

Code	**Omschrijving**		**Aantal**	**Totaal**
() C85	Weekendbehandeling	21,78	1	21,78
() C86	Avondbehandeling	21,78		
() C87	Nachtbehandeling	21,78		
() C13	Probleemgericht consult 47	21,78	1	21,78
() E01	Wortelkanaalbehandeling consult	21,78		
() E02	Uitgebreid wortelkanaalconsult	40,12		
() E03	Consult na een tandheelkundig ongeval	31,52		
() E77	Initiele wortelkanaalbehandeling 1e kanaal	57,31		
() E78	Initiele wortelkanaalbehandeling elk volgend kanaal	28,66		
() X10	Kleine rontgenfoto	16,05	1	16,05
() H11	Trekken tand of kies	42,98		
() H35	Moeizaam trekken tand of kies	68,77		
() V91	Eenvlaksvulling	45,85		
() V92	Tweevlaksvulling	60,18		
() V93	Drievlaksvulling	71,64		
() A10	Verdoving	14,33		
() R74	Opnieuw vastzetten kroon	22,92		
()				
()				
()				
()				

[illegible]
15-09-2019

Totaal 59,61

Stempel tandartsenpraktijk
Tandartsenpraktijk DENTIS
tandarts: A.P. Uriot
Boermarkeweg 54
7824 AA Emmen
tel:0591-617618

(c) Example of an image with high std of the regional skew angles.

Figure 7.10: Examples of images with and without skew.

We also try splitting the image into 6 regions: 3 by vertical and 2 by horizontal and apply CV, PP, and the Hough methods to these regions as an alternative to the histogram features. This results in 6 new features. However, the use of these features instead of the histogram features gave worse classification results during the experiments and, hence, they are not investigated further.

Let's look into several examples and estimate the quality of the suggested skew evaluation methods. We look at the images with different visible levels of skew. The observed results are presented in Table 7.1. We can confirm that the Hough method is the least reliable as mentioned before, while 4 other methods may sometimes give incorrect results. For example, CV method for Figure 7.10b or both Sauvola and CV methods for Figure 7.10c provide inaccurate skew angle estimates.

Thus, we have seen that skew evaluation methods perform differently. Some of them do not provide reliable results. Others perform well on the majority of the images but can still sometimes give incorrect results. Therefore, in the classification algorithm, we will use the skew features extracted using several different methods.

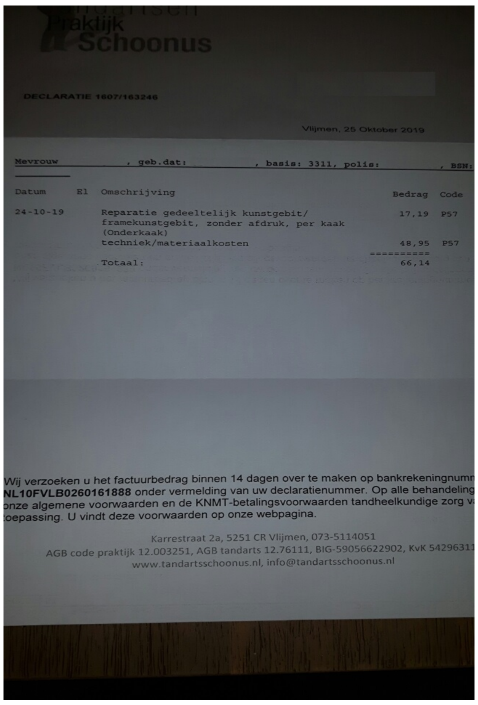
7.3.2 Other text features

Other text features which are important for recognition algorithm are the font of the text and its closeness to the borders of the image. The latter helps to understand if the text is not fully seen.

In order to evaluate font quality, a font thickness metric is developed. Firstly, the image is cropped by 10% from each side, downscaled by the factor of 2, and binarized. Then, the bursting procedure from [43] is applied in the horizontal direction, transforming each sequence of black pixels into one pixel with a value of the sequence width. Then, we split the image into the blocks of size 32×32 and calculate the mean width and the standard deviation of the widths in each block. Finally, we use the mean over all blocks and the histograms of the mean and standard deviation values with 8 equidistant bins in the interval $[0, 4]$ as features. We consider only this interval as large font thickness usually does not pose any problems for recognition. Besides this, we also use the mean of the values which lie in the interval $[0, 4]$ as a feature. This helps to omit large values which can also correspond to the horizontal paper edges or some other horizontal non-text elements.

These features do not always represent font thickness as they are supposed to: there are a lot of additional lines on images whose thickness is also considered as font thickness. Besides this, some letters also have horizontal strokes. However, in many cases, this metric represents font thickness well. For example, in Figure 7.11 the images with a thin and thick fonts are presented.

To estimate the distance from the text to the borders we also have to binarize the image. After that, this distance can be estimated in several ways. The first, simple way, is just to find the closest black pixel to each of the borders of the image. The second approach is to apply the bursting procedure in the horizontal direction both from the left to the right and the other way around, and in the vertical direction from the bottom to the top. This way, we can find the right-most, the left-most, and the top-most pixels of the lines in the image. Then, we find the closest pixel to the border with a value in the pre-specified interval $[2, 10]$ for horizontal lines and $[20, 50]$ for vertical lines which correspond to the possible font thickness and letter height respectively. The distance from this pixel to the respective border is considered the


Praktijk
Schoonus

[redacted]

DECLARATIE 1607/163246

Vlijmen, 25 Oktober 2019

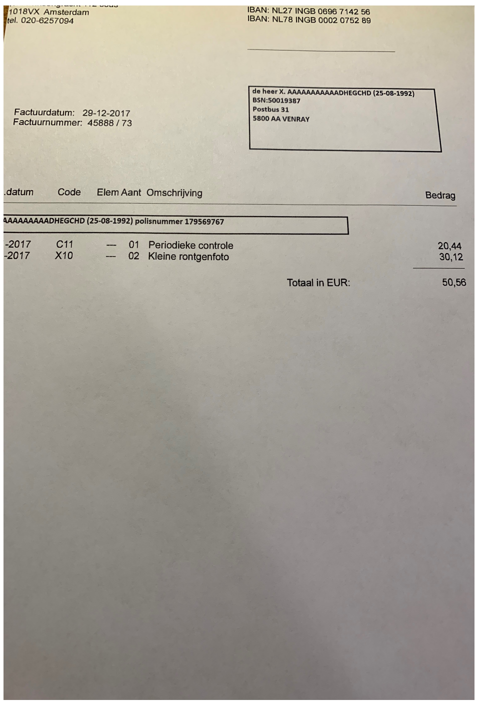
Mevrouw , geb.dat: , basis: 3311, polis: , BSN:

Datum	El	Omschrijving	Bedrag	Code
24-10-19		Reparatie gedeeltelijk kunstgebit/ framekunstgebit, zonder afdruk, per kaak (Onderkaak)	17,19	P57
		techniek/materiaalkosten	48,95	P57
		Totaal:	66,14	

Wij verzoeken u het factuurbedrag binnen 14 dagen over te maken op bankrekeningnumm
NL10FVLB0260161888 onder vermelding van uw declaratienummer. Op alle behandeling
onze algemene voorwaarden en de KNMT-betalingsvoorwaarden tandheelkundige zorg v
toepassing. U vindt deze voorwaarden op onze webpagina.

Karrestraat 2a, 5251 CR Vlijmen, 073-5114051
AGB code praktijk 12.003251, AGB tandarts 12.76111, BIG-59056622902, KvK 5429631
www.tandartsschoonus.nl, info@tandartsschoonus.nl

(a) Example of an image with low font thickness equal to 1.7.


1018VX Amsterdam
tel. 020-6257094

IBAN: NL27 INGB 0696 7142 56
IBAN: NL78 INGB 0002 0752 89

de heer X. AAAAAAAAAAADHEGCHD (25-08-1992)
BSN:50019387
Postbus 31
5800 AA VENRAY

Factuurdatum: 29-12-2017
Factuurnummer: 45888 / 73

.datum	Code	Elem	Aant	Omschrijving	Bedrag
AAAAAAAAADHEGCHD (25-08-1992) polisnummer 179569767					
-2017	C11	---	01	Periodieke controle	20,44
-2017	X10	---	02	Kleine rontgenfoto	30,12
				Totaal in EUR:	50,56

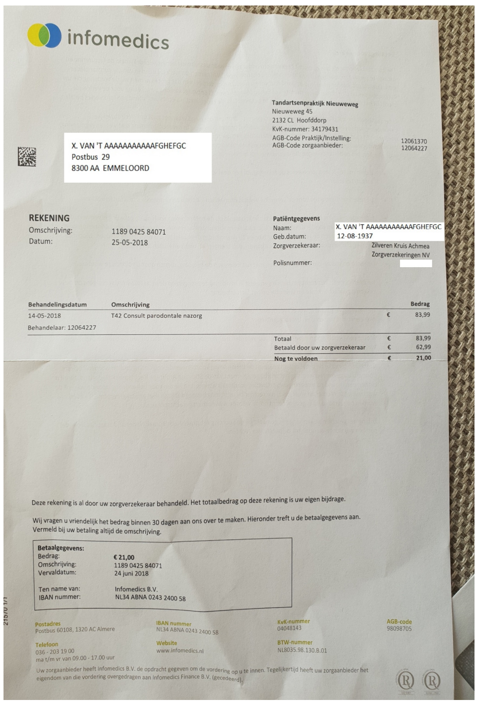
(b) Example of an image with high font thickness equal to 7.1.

Figure 7.11: Examples of images with high and low font thickness.

distance from the text to this border. This way, we decrease the influence of noise and edges of the paper on the distance metric. However, this metric is not accurate due to the presence of various non-text elements and not precise estimation of the intervals for the pixels values which to consider part of the text.

That is why another possible method for estimating the distance from the text to the border is implemented, too. In this method, we detect text regions first and then calculate the distance from them to the borders of an image. This can be done by applying morphological operations mentioned in Section 4.3.1 and explained in Section 7.3.1. Firstly, we apply a closing operator with a line structuring element of size 50×1 and then, an opening with a structuring element of size 10×10. Then, we measure the distance from the detected text regions to the left, right, and top borders. Besides this, we crop the image by $3/8$ from above and below in order to measure the distance from the left and the right borders in the middle of the image. This way we reduce the impact of non-text elements which are located in the top and bottom parts of the image.

The morphological method is bad in distinguishing background texture (if present on an image) from the text. Thus, paper edges and the surface on which it lies decreases the precision of the text localization and, hence, distance measurements are not correct. For example, for the image in Figure 7.12a the estimated distance to the right border equals 0 because of the textured background present on the right side of the image. On the other hand, for the image in Figure 7.12b, the distance to the right border is 388 which is close to reality.



infomedics

Tandartsenpraktijk Nieuweweg
Nieuweweg 45
2132 CL Hoofddorp
KvK-nummer: 34179431
AGB-Code Praktijk/Instelling: 12061370
AGB-Code zorgaanbieder: 12064227

X. VAN 'T AAAAAAAAAAAFGHEFGC
Postbus 29
8300 AA EMMELOORD

REKENING
Omschrijving: 1189 0425 84071
Datum: 25-05-2018

Patiëntgegevens
Naam: X. VAN 'T AAAAAAAAAAAFGHEFGC
Geb.datum: 12-08-1937
Zorgverzekeraar: Zilveren Kruis Achmea Zorgverzekeringen NV
Polisnummer:

Behandelingsdatum	Omschrijving		Bedrag
14-05-2018	T42 Consult parodontale nazorg	€	83,99
Behandelaar: 12064227			
	Totaal	€	83,99
	Betaald door uw zorgverzekeraar	€	62,99
	Nog te voldoen	€	**21,00**

Deze rekening is al door uw zorgverzekeraar behandeld. Het totaalbedrag op deze rekening is uw eigen bijdrage.

Wij vragen u vriendelijk het bedrag binnen 30 dagen aan ons over te maken. Hieronder treft u de betaalgegevens aan. Vermeld bij uw betaling altijd de omschrijving.

Betaalgegevens:
Bedrag: **€ 21,00**
Omschrijving: 1189 0425 84071
Vervaldatum: 24 juni 2018

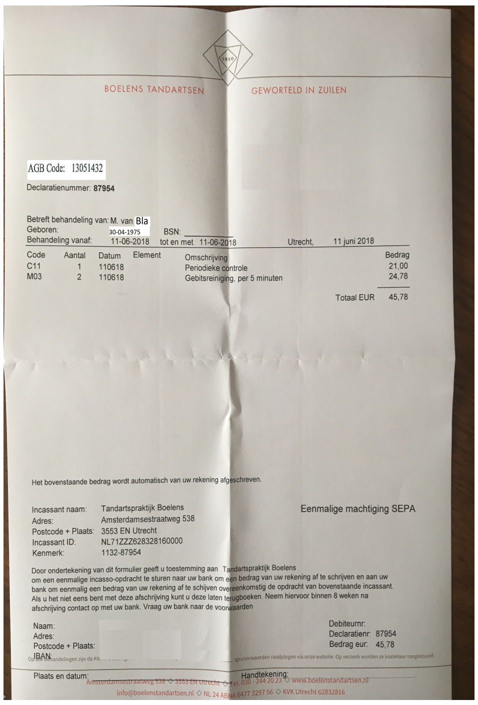
Ten name van: Infomedics B.V.
IBAN nummer: NL34 ABNA 0243 2400 58

215/0 1/1

Postadres Postbus 60108, 1320 AC Almere
IBAN nummer NL34 ABNA 0243 2400 58
KvK-nummer 04048143
AGB-code 98098705
Telefoon 036 - 203 19 00 ma t/m vr van 09.00 - 17.00 uur
Website www.infomedics.nl
BTW-nummer NL8035.98.130.B.01

Uw zorgaanbieder heeft Infomedics B.V. de opdracht gegeven om de vordering op u te innen. Tegelijkertijd heeft uw zorgaanbieder het eigendom van die vordering overgedragen aan Infomedics Finance B.V. (gecedeerd).

(a) Example of an image with the distance to the right border equal to 0.



1950

BOELENS TANDARTSEN GEWORTELD IN ZUILEN

AGB Code: 13051432
Declaratienummer: **87954**

Betreft behandeling van: M. van Bla
Geboren: 30-04-1975 BSN:
Behandeling vanaf: 11-06-2018 tot en met 11-06-2018 Utrecht, 11 juni 2018

Code	Aantal	Datum	Element	Omschrijving	Bedrag
C11	1	110618		Periodieke controle	21,00
M03	2	110618		Gebitsreiniging, per 5 minuten	24,78
				Totaal EUR	45,78

Het bovenstaande bedrag wordt automatisch van uw rekening afgeschreven.

Eenmalige machtiging SEPA

Incassant naam: Tandartspraktijk Boelens
Adres: Amsterdamsestraatweg 538
Postcode + Plaats: 3553 EN Utrecht
Incassant ID: NL71ZZZ628328160000
Kenmerk: 1132-87954

Door ondertekening van dit formulier geeft u toestemming aan Tandartspraktijk Boelens om een eenmalige incasso-opdracht te sturen naar uw bank om een bedrag van uw rekening af te schrijven en aan uw bank om eenmalig een bedrag van uw rekening af te schijven overeenkomstig de opdracht van bovenstaande incassant. Als u het niet eens bent met deze afschrijving kunt u deze laten terugboeken. Neem hiervoor binnen 8 weken na afschrijving contact op met uw bank. Vraag uw bank naar de voorwaarden

Naam:
Adres:
Postcode + Plaats:
IBAN:

Debiteurnr:
Declaratienr: 87954
Bedrag eur: 45,78

Op alle behandelingen zijn de KN [illegible] ...gsvoorwaarden raadplegen via onze website. Op verzoek worden ze kosteloos toegestuurd

Plaats en datum: Handtekening:

Amsterdamsestraatweg 538 ◇ 3553 EN Utrecht ◇ Tel. 030 - 244 20 23 ◇ www.boelenstandartsen.nl
info@boelenstandartsen.nl ◇ NL 24 ABNA 0477 3297 56 ◇ KVK Utrecht 62832816

(b) Example of an image with the distance to the right border equal to 388.

Figure 7.12: Examples of the images with different distance between the text and the right border calculated using the morphological method.

Thus, in this section, we have suggested several text-related features such as font thickness and distance from the text to the borders of the image. These features do not always accurately represent what they are supposed to. However, in many cases, they are good estimates of the respective quantities. Therefore, we keep these features for further analysis.

7.4 PCA features

As one more source of features, we can consider the features extracted from images by means of PCA. We convert images to the grayscale, resize all of them to the standard size and flatten. Then, we perform PCA analysis on this data in order to get components that explain the largest share of the variance and keep them as features. It turned out that there exists one component explaining more than 50% of the variance and the first 50 components together explain 85% of the variance. The feature which explains the most of the variance turns out to be highly correlated with the ratio contrast feature with the estimated correlation coefficient of 0.93. At the same time, all the other features do not have the correlation higher than 0.8 with already extracted features. However, these PCA features do not add value to the classification model and do not provide sufficiently high precision and recall when they are used alone for classification. Thus, we do not investigate these features further.

Feature \ Image size	1200 × 900		2500 × 1900		4000 × 3000	
	mean	std	mean	std	mean	std
BRISQUE (5 feature sets)	0.287	0.031	1.186	0.152	2.893	0.291
IL-NIQE gradient (3 feature pairs)	0.509	0.032	2.226	0.183	5.578	0.242
IL-NIQE color	0.289	0.021	1.263	0.106	3.167	0.114
IL-NIQE simple Gabor (25 feature pairs)	5.551	0.549	25.822	0.783	64.753	0.685
IL-NIQE gradient Gabor (25 feature pairs)	5.801	0.554	26.911	0.736	67.283	5.874
Size	0.024	0.005	0.103	0.015	0.251	0.035
Average brightness	0.031	0.004	0.137	0.006	0.347	0.012
Max brightness	0.018	0.004	0.077	0.005	0.193	0.009
Luma brightness	0.031	0.005	0.137	0.006	0.347	0.014
Luminance brightness	0.031	0.004	0.136	0.005	0.348	0.013
Contrast ratio (2 features)	0.002	0.0005	0.010	0.001	0.065	0.002
Contrast RMS	0.014	0.002	0.062	0.004	0.157	0.008
Hue and saturation	0.005	0.001	0.019	0.002	0.047	0.003
Laplace sharpness	0.028	0.003	0.122	0.006	0.312	0.012
S3 spectral sharpness	0.120	0.021	0.549	0.022	1.378	0.045
S3 spatial sharpness	0.112	0.039	0.480	0.036	1.220	0.084
S-index	0.534	0.056	2.533	0.092	6.389	0.177
FISH sharpness	0.048	0.009	0.255	0.031	0.691	0.068
Regional Laplace sharpness	0.015	0.003	0.070	0.134	0.166	0.026
Regional S3 spatial sharpness	0.050	0.008	0.225	0.041	0.565	0.090
Font thickness	0.020	0.005	0.069	0.010	0.189	0.028
Border distance	0.121	0.019	0.543	0.052	1.520	0.122
Borders morphology	0.249	0.039	1.080	0.126	2.711	0.296
Hough orientation	0.044	0.017	0.153	0.057	0.354	0.135
Morphology orientation	0.038	0.015	0.324	0.143	1.521	0.771
PP orientation	0.065	0.020	0.229	0.070	0.513	0.157
CV orientation	0.060	0.020	0.253	0.063	0.582	0.147
Sauvola orientation	0.003	0.001	0.012	0.003	0.029	0.005

Table 7.2: Computational complexity of the feature extraction methods (in seconds) for different image sizes based on the analysis of 370 images.

7.5 Computational complexity analysis of feature extraction

In this section, we consider the computational complexity of the feature extraction algorithms. Images were resized to the specific sizes which often appear in the dataset: 1200 × 900, 2500 × 1900, and 4000 × 3000. For each of the sizes, the feature extraction algorithms were timed only on 370 different images as this process takes a long time to run for all the images. The results can be found in Table 7.2.

We can see that IL-NIQE features and S-index are the most computationally expensive features. Other features whose extraction is also slow for large images are border features, S3 sharpness, and morphological orientation features. Thus, it would be good if we can omit some of these features and if computing of the remaining ones on downscaled images does not deteriorate the quality of the classification algorithm. We will analyze these possibilities in Sections 8.2.3 and 8.2.4.

Chapter 8

Recognition prediction

8.1 Classification algorithms description

In this section, we formulate the classification problem and discuss machine learning algorithms that can be used to solve it.

To begin with, let's formulate the classification problem. There are two classification problems considered in out research:

- Classification with all 3 classes: positive, partial and negative;
- Classification with 2 classes: positive and negative:
  - Partial class included in the negative class;
  - Partial class included in the positive class.

We assign the following numerical labels to the classes in the 3 classes classification: 0 for the negative class, 1 — for partial, and 2 — for positive.

When the classification into 3 classes is performed, we can also introduce order of the classes as the positive, partial, and negative classes can be considered as image quality estimates. Then, partially recognized images are somewhere between the negative and positive classes in terms of the image quality. Thus, we can use some regression algorithm for the classification, assigning images to the classes according to the obtained values. To do so, we have to introduce the thresholds, which will allow us to choose the class according to the obtained value. The first threshold is a value in $[0, 1]$ that splits the negative and the partial classes. Another threshold is a value in $[1, 2]$ that splits the partial and the positive classes. We will refer to these thresholds as to "partial" and "positive" thresholds and will use the notation $a|b$, where a is a partial threshold and b is a positive one.

Furthermore, we can also use a threshold when classifying into 2 classes. Then, it will be only one probability threshold which takes values in $[0, 1]$. Thus, if the probability of an image being assigned to the positive class is larger than this threshold, we classify it as positive and, otherwise, as negative.

The next step is to choose a machine learning algorithm for these classification problems. For classification, we use methods from the sklearn package in Python [3]. We have tried using random forest classification and SVM with the default hyper-parameters for both classification problems: into 2 classes and 3 classes. SVM failed to perform the classification, resulting in the same prediction for all samples while random forest gave better results. Therefore, we use only random forest models in further research.

Two types of random forests are used: a classifier and a regressor. The random forest regressor is used for the ordered classification into 3 classes as the classifier does not account for ordering. The random forest classifier is used for classification both into 2 and 3 classes. We also explore the possibility of using a random forest regressor for classification into 2 classes, although, it should not behave much differently from a random forest classifier.

Let's discuss how these models work. We introduce the notation $\{(X_m, y_m)\}_{m=1}^{M}$ for the dataset, where $X_m = (x_1, x_2, \ldots, x_k)_m$ is the set of features (independent variables) and y_m is the target value (dependent variable) for the sample m. y_m refers to the class in the classification model and to some real number in the regression model. We will discuss the random forest models using these notations.

Random forest is an ensemble of decision trees, classification models described in [92]. Each of the decision trees is constructed using only a subset $\{\bar{X}_m\}_{m=1}^{M}$ of the features which is selected randomly. The size of $\bar{X}_m$ is one of the random forest parameters. Often, it equals to the square root of the total number of features. When the subset is chosen, a decision tree is constructed. All the data points are firstly located in the root node. Then, at each node n, the best feature to split the data is chosen according to some criterion. There are different criteria for random forest classification and regression models. Let's start with the criterion used in the classification model. There, we use the Gini impurity $I_G(n)$ criterion (eq. 8.1). According to it, the feature that results in a split with the largest reduction of the Gini impurity is selected to perform this split. This reduction for a vertex n is calculated as difference of $I_G(n)$ and the weighted sum of the Gini impurity of children of the vertex n (eq. 8.2).

$$I_G(n) = 1 - \sum_{i=1}^{J} p_i^2, \tag{8.1}$$

where J is the number of classes and p_i is the number of samples from class i in the node n.

$$\mathrm{red}(I_G) = I_G(n) - \frac{|n_r|}{|n|} I_G(n_r) - \frac{|n_l|}{|n|} I_G(n_l), \tag{8.2}$$

where n_r and n_l are the children nodes of the node n, and $|n|$, $|n_r|$, and $|n_l|$ are the numbers of samples falling into the respective nodes.

In the regression model, the criterion is the largest reduction of mean squared error (MSE) which is given in eq. 8.3.

$$\mathrm{red}(MSE) = \sum_{i \in n} (y_i - y_n^*)^2 - \sum_{i \in n_l} (y_i - y_{n_l}^*)^2 - \sum_{i \in n_r} (y_i - y_{n_r}^*)^2, \tag{8.3}$$

where y_n^*, $y_{n_l}^*$, and $y_{n_r}^*$ are the means of target values y_i in the respective nodes.

Some other hyper-parameters of the model are the maximum allowed depth of the trees, the minimum number of samples in the node for splitting it, and the minimum number of samples in a leaf. These parameters help to avoid overfitting as if we perform too many splits, we can end up with leaves with single samples.

The splitting process is stopped when there is no reduction of the criterion can be obtained, the maximum depth is reached, or the number of samples in a node or resulting leaf after a split does not satisfy the hyper-parameters mentioned above.

In the end, each leaf of the decision tree has some samples. Calculations of the final prediction of a decision tree differ for a classification and regression decision trees. In the classification model, the samples in the leaf belong to different classes and, thus, provide an empirical distribution over these classes. When a test sample is passed through the decision tree, it returns this empirical distribution over classes and the class with the highest probability is considered as a predicted class for the test sample.

In the regression model, the samples in the leaves have some target values y_i and the prediction of the tree is the average of the values in the leaf.

The outcome of the random forest classifier for a test sample is the probability distribution over the classes that is calculated as the mean of the empirical distributions over all decision trees. The prediction of the random forest regressor is the average of the predictions over all decision trees.

Thus, in this section, we have formulated the classification problems with 2 and 3 classes. Furthermore, we have discussed the random forest models which we will use for classification in this project. Throughout the report, we will use the following order of the random forest parameters: the number of estimators (trees), the maximum depth, the minimum number of samples to split the node, the minimum number of samples in a leaf, the sizes of a features subset to consider for splitting a node ("auto" corresponds to the square root of the number of features).

8.2 Evaluation and comparison of the models

After choosing the classification algorithms, we evaluate them and compare their performance with different hyper-parameters, different number of features, and on different data. In this section, we elaborate on the methods and results of evaluation and comparison.

8.2.1 Evaluation methods

To begin with, we discuss the evaluation methods that we use. We use several metrics for evaluation: classification accuracy (eq. 8.4), positive precision (referred to as positive prediction value in the literature, eq. 8.5) and negative precision (referred to as negative predictive value in the literature, eq. 8.6).

$$acc = \frac{\#\text{ correctly classified images}}{\#\text{ all images}}. \tag{8.4}$$

$$prec_{\text{pos}} = \frac{\#\text{ correctly classified positive images}}{\#\text{ all images classified as positive}}. \tag{8.5}$$

$$prec_{\text{neg}} = \frac{\#\text{ correctly classified negative images}}{\#\text{ all images classified as negative}}. \tag{8.6}$$

As we have more positive examples than negative and partial ones (Section 5), accuracy might be not fully representative of the quality of the classification method. That is why we also consider 2 precision metrics. We can give preference to one of them depending on our final goal of classification. Originally, we need classification in order to decide if we have to ask a user for a new photo immediately or whether we should pass the image through the ORTEC algorithm first. Thus, the default action is to pass the image to the ORTEC algorithm. If we want to pass it only if we are sure with some high probability that the image will be eventually recognized, we should focus on positive precision which can be interpreted as the probability of the image being recognized by the ORTEC algorithm if it is classified as positive. On the other hand, if we are interested in asking the user for a new photo only if we are sure with a high probability that it wouldn't be recognized by the ORTEC algorithm we should focus on negative precision. It was decided that it is more important to ask for a new photo only if we are sure that it is needed, hence, the focus should be made on the negative precision. We will start by reporting both precision values and then focus on the negative precision.

Alongside with the negative precision, we will also refer to the negative recall (referred to as true negative rate in the literature, eq. 8.7). It will be useful when we select the threshold and the features for the model as it is also important to know which percentage of the negative examples are actually classified as negative by the classification model. Positive recall (referred to as true positive rate in the literature) can be defined analogically but it is not used in this project.

$$recall_{\text{neg}} = \frac{\#\text{ correctly classified negative images}}{\#\text{ all negative images}}. \tag{8.7}$$

All of the mentioned metrics are calculated using k-fold cross-validation. This way, we obtain a prediction for each image and, hence, can calculate these metrics over the whole dataset.

To choose the number of folds k, we run a loop over different values of $k \in \{2, 3, \ldots, 20\}$ and plot the graph (Fig. 8.1) of metrics depending on k for the random forest classification model with 2 classes including partial into negative, parameters $(200, 20, 3, 2, \text{auto})$, and the threshold 0.5. We can see that the number of folds does not influence the classification algorithm quality too much for $k > 5$ and, as we would like to choose k, that will be reasonable in terms of both computational complexity and evaluation quality, we can consider $k = 10$ in the further analysis. The graphs for models with other thresholds and parameters have similar form and, hence, $k = 10$ will suit for analysis of any model. We will report all the evaluation metrics as the mean of the metrics over 20 simulations and will also mention the standard deviation.

As a baseline, we use classification, where all the images are classified as positive. This is the default behaviour which resembles the way how the current ORTEC algorithm works as all the images despite of the quality are directly passed to the OCR and the ORTEC algorithm.

Thus, we have discussed the evaluation methods that are used in the project and have explained our choices. In the next sections, we will compare different models using these evaluation metrics.

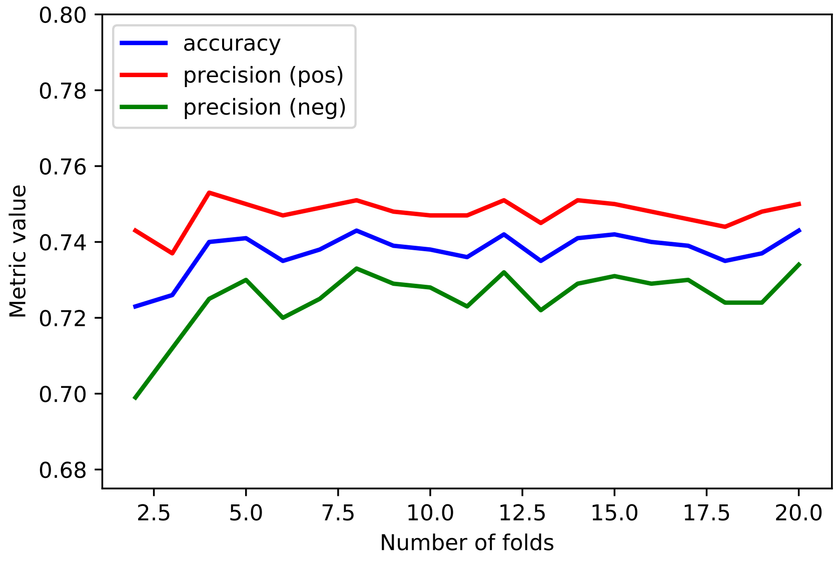


Figure 8.1: Metrics values depending on the number of folds k in k-fold cross-validation.

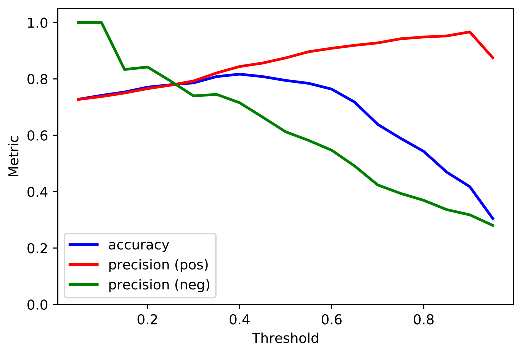
8.2.2 Classification algorithms comparison and evaluation

Now, when we have introduced the classification models and the evaluation methods for them, we perform comparisons of different models. As discussed in Chapter 5, we have 584 images in the dataset: 312 in the positive class, 112 in the partial, 160 in the negative. Moreover, we use the features discussed in Sections 7.1, 7.2 and 7.3. There are 401 features that are at first used for classification.

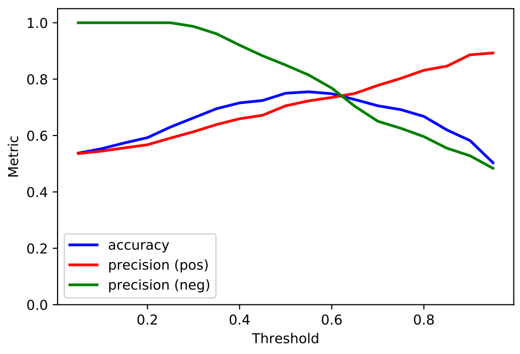
2 classes classification

Let's start by comparing several models for classification into 2 classes. Firstly, we have to decide, into which class, positive or negative, to include partial examples. Let's first use a random forest regressor with parameters $(200, 20, 3, 2, \text{auto})$ for classification into 2 classes for both options and try different threshold values to distinguish positive instances from negative. Threshold values vary between 0 and 1 with a step of 0.05. In Figures 8.2a- 8.2b, we can see the 3 metrics depending on the threshold value. We can see, that when partial examples are considered as negative, the negative precision is higher than in the case when we consider them as positive examples. At the same time, the accuracy and positive precision are slightly larger in another case. As we are interested in the negative precision, we have decided that in the case of random forest regressor for 2 classes, we will work with the model where partial examples are considered as negative. However, we could have chosen another option if we had been focused on the positive precision.

Now, let's look at the random forest classifier model with 2 classes with the same parameters. The accuracy is similar in both models in this case, so we focus more on the plots of the positive precision and recall (Fig. 8.3) and of the negative precision and recall (Fig. 8.4).

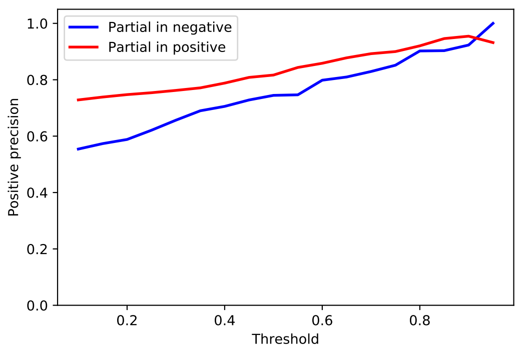


(a) Evaluation metrics with partial examples classified as positive.

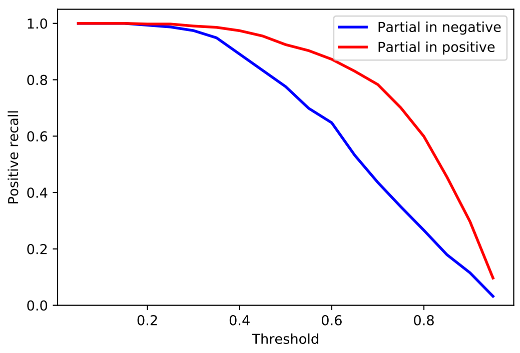


(b) Evaluation metrics with partial examples classified as negative.

Figure 8.2: Evaluation metrics for the random forest regressor with 2 classes.

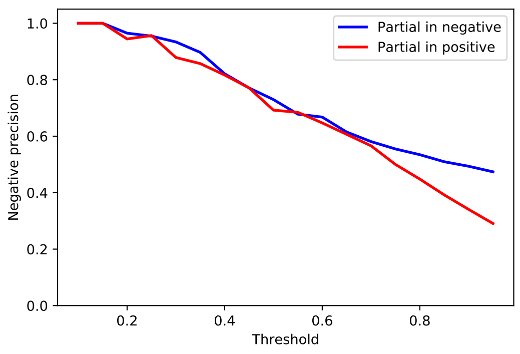


(a) Positive precision.

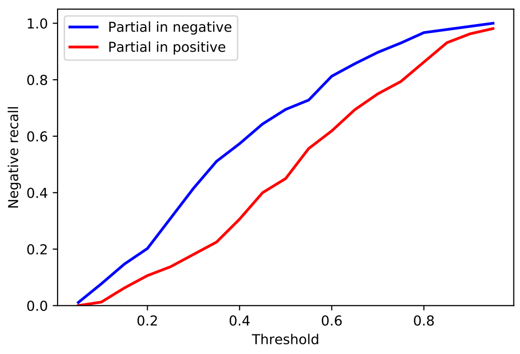


(b) Positive recall.

Figure 8.3: Positive precision and recall for the random forest classifier with 2 classes.



(a) Negative precision.



(b) Negative recall.

Figure 8.4: Negative precision and recall for the random forest classifier with 2 classes.

We can see that the positive precision and recall are almost always higher when we include partial examples in the positive class. On the other hand, the negative recall is always higher when we include partial examples into the negative class while the negative precision is also slightly higher in this case. Thus, based on these graphs, we can conclude that we should include partial examples into positive class if we focus on the positive precision and recall and into the negative class if we focus on the negative precision and recall. Thus, in this project, we include partial examples into negatives for the random forest classifier, too. This choice also has a common-sense explanation as the partial class consists of the incorrectly partially recognized images which means that there is some information present in the images that is not recognized by OCR. Thus, in this sense, the partial and negative classes are similar with the difference that there is less not recognized information in the partial class compared to the negative.

Furthermore, we can choose thresholds for the models which maximize the metrics we are interested in. Based on Fig. 8.2b and Fig. 8.4, we decide to look at the thresholds of 0.2, 0.3 and 0.4 for which the high values of the negative precision are obtained and the negative recall values are not extremely low. Depending on the needs of the business, one of these thresholds can be chosen in the final model. Besides this, we will sometimes still consider the threshold of 0.5 as it maximizes the accuracy.

Therefore, we have decided to include the partial class in the negative class when performing classification into 2 classes. Moreover, we have chosen several threshold values to consider for classification into 2 classes.

Classification methods comparison

In this subsection, we will compare random forest regression and classification models with different numbers of classes, parameters, and thresholds. For the classification, we use 189 features out of 401, retaining only the uncorrelated features (it is discussed in Section 8.2.3). For random forests, presence of unimportant features among these 189 does not deteriorate the model so we do not consider removing them in this section.

In Table 8.1 the accuracy, positive and negative precision, and negative recall are given for random forest classifier (RFC) and regressor (RFR) with 2 and 3 classes with various thresholds and parameters. The parameters and thresholds were chosen by grid-search on the accuracy, positive precision, and negative precision. Some parameter sets were used on multiple models to allow direct comparisons of performance. Besides this, the number of estimators was manually decreased sometimes in order to see if it significantly influences the performance of the model or not, so that we can decrease the computational complexity of evaluation procedures without losing much in the quality of performance. The parameters can still be suboptimal as the grid-search did not explore a lot of parameter sets as it will be very computationally expensive.

We can see, that depending on the metric which we want to maximize, we should choose different parameters and thresholds. As we mentioned in Section 8.2.1, we focus mostly on the negative precision and negative recall. For maximizing these metrics the models with 2 classes work better. We can see that for the random forest classifier with 2 classes the highest negative precision is obtained in the model with the parameters $(400, 15, 3, 2, \text{auto})$ and a threshold of 0.3. The negative recall for this model is not much lower compared to other random forest classifiers with negative precision of around 0.9. Thus, we would often use this

Model	Threshold	Parameters	Accuracy		Precision pos.		Precision neg.		Recall neg.	
			mean	std	mean	std	mean	std	mean	std
Baseline			0.534		0.534					
RFC. 2 cl.	0.5	400, 40, 4, 3, 0.8	0.735	0.006	0.741	0.004	0.727	0.008	0.69	0.005
RFC. 2 cl.	0.8	200, 20, 3, 3, 0.5	0.608	0.008	**0.871**	0.016	0.546	0.005	**0.947**	0.007
RFC. 2 cl.	0.3	200, 20, 3, 3, 0.5	0.719	0.003	0.664	0.002	0.905	0.011	0.443	0.006
RFC. 2 cl.	0.3	400, 40, 4, 3, 0.8	0.72	0.005	0.666	0.004	0.896	0.009	0.45	0.011
RFC. 2 cl.	0.3	400, 15, 3, 2, auto	0.716	0.006	0.657	0.004	0.939	0.008	0.416	0.01
RFC. 2 cl.	0.3	200, 15, 3, 2, auto	0.711	0.006	0.655	0.004	0.929	0.011	0.412	0.01
RFC. 3 cl.		600, 15, 3, 3, auto	0.673	0.006	0.695	0.004	0.651	0.007	0.624	0.013
RFR. 3 cl.	0.5 — 1.5	600, 15, 3, 3, auto	0.494	0.003	0.81	0.006	0.769	0.017	0.254	0.007
RFR. 3 cl.	0.9 — 1.5	200, 25, 3, 2, 0.6	0.547	0.007	0.807	0.008	0.664	0.013	0.609	0.01
RFR. 3 cl.	0.65 — 1.6	200, 20, 3, 2, 0.8	0.474	0.007	0.83	0.01	0.742	0.008	0.4	0.01
RFR. 2 cl.	0.5	200, 20, 3, 2, 0.8	**0.746**	0.005	0.701	0.004	0.851	0.009	0.551	0.007
RFR. 2 cl.	0.3	400, 15, 3, 2, auto	0.661	0.005	0.612	0.003	**0.986**	0.004	0.276	0.01

Table 8.1: Evaluation metrics for different random forest models.

set of parameters further in the project. Sometimes, we will choose $(200, 15, 3, 2, \text{auto})$ if we will need to reduce the time spent on computations as the negative precision and recall do not decrease significantly in this case compared to $(400, 15, 3, 2, \text{auto})$. We also obtain high values of the negative precision in the random forest regressor with 2 classes, though we still have to tune it in order to obtain relatively high negative recall, too. For classification into 2 classes, it does not matter much if we use classifier or regressor, so we choose to use the random forest classifier.

Thus, we have decided to focus on classification into 2 classes. For it, we have chosen the random forest classifier models with the parameters $(400, 15, 3, 2, \text{auto})$ or $(200, 15, 3, 2, \text{auto})$ and the thresholds 0.2, 0.3, and 0.4 which were mentioned in the previous subsection about 2 classes classification.

8.2.3 Feature selection

We have mentioned in Section 8.2.2 that there are 401 features in total. Some of them are correlated with each other and some of them are unimportant for the classification. In this section, we perform feature selection in order to reduce the size of the feature set and, hence, decrease the computational complexity of the feature extraction part.

Feature selection is performed in several steps. The first step is to calculate correlation between the features and for those pairs where the correlation is large (larger than 0.8) one of the features in the pair is removed. Afterwards, the features importance is calculated in the chosen classification algorithm using different feature importance calculation algorithms. Finally, the mean and the variance of the importance values are used to choose and remove the irrelevant features.

Correlated features

Firstly, we remove correlated features so that we do not have highly correlated ones in our features set. Out of 401 original features, 189 are not highly correlated and are kept for further analysis. These features are the following:

- 10 BRISQUE features:
  - parameters ν and σ for mscn coefficients;
  - parameters ν and σ_l for one of the orientations (any can be chosen as they are correlated);
  - the mean and σ_r for horizontal, vertical and one of the diagonal orientations;
- 26 IL-NIQE features:
  - 4 horizontal gradient IL-NIQE features: parameters of GGD fitted to 2 channels O_1 and O_2;
  - All 6 IL-NIQE color features;
  - 11 simple Gabor features:
    * a pair of parameters for the orientation $\pi/5$ and scale $1/4$;
    * 5 parameters for the orientation 0;
    * 4 parameters for the orientation $2\pi/5$;
  - 5 horizontal gradient Gabor features:
    * 3 parameters for the orientation 0;
    * 1 parameter for the orientation $\pi/5$ and scale $1/2$;
    * 1 parameter for the orientation $3\pi/5$ and scale $1/8$;
- 2 image dimensions parameters;
- Image size in bytes;
- 1 brightness metric;
- 3 contrast metrics (calculated for the chosen brightness metric);
- Hue and saturation parameters;
- Spectral and spatial S3 sharpness;
- S-index;
- 44 histogram values for regional Laplace sharpness and the mean of large regional sharpness values;
- 6 histogram values for regional S3 spatial sharpness metric and the standard deviation, minimum and maximum values;
- 15 histogram values for font thickness, the mean and the mean of small values;

- All 12 border features: usual and morphological;
- Mean for morphological and PP skew evaluation methods and 5 other skew angle estimates: 2 estimates from the Hough and one from CV, PP, and Sauvola methods;
- All 51 skew histogram values;

Thus, we can already omit some calculations for feature extraction. For example, we omit calculation of BRISQUE for the second diagonal, calculations of IL-NIQE for vertical gradient image, most of the orientations and scales for log-Gabor transform of original and horizontal gradient images, all but one brightness and related contrast metrics and standard deviation and maximum for regional orientation calculation methods.

Feature importance calculation algorithms

Now, we calculate feature importance values which will allow us to remove unimportant features from the remaining ones.

The algorithms for calculation of feature importance which can be used are the permutation feature importance algorithm ([3]), Shapley feature importance, and feature importance calculation algorithm provided by the random forest classification algorithm.

Permutation feature importance for feature i is calculated using the following scheme:

1. Score is calculated on the original data;
2. Values of the feature i are permuted and the score is evaluated again;
3. Difference in the scores is considered as feature importance.

As a score, different evaluation metrics can be used such as accuracy or precision.

Shapley value ϕ is the notion from game theory that can be also used to calculate feature importance as discussed in [24]. It represents the marginal contribution of the feature to all possible features combinations. Formally, it is defined as in eq. 8.8.

$$\phi(i) = \frac{1}{\Pi} \sum_{\pi \in \Pi} \Delta_i(S_i(\pi)), \tag{8.8}$$

where Π is the set of random permutations, $S_i(\pi)$ is the set of features which appear before i feature in permutation π and $\Delta_i(S)$ defines marginal importance of feature i to set S and in the case of feature importance equals to difference in predictions. Averaging over all permutations is a very computationally expensive procedure. Thus, an approximation of Shapley importance can be calculated as the average over permutations of size $\sqrt{N}$, where N is the total number of features.

In a random forest classifier, a so-called Gini importance criterion is used for feature importance calculation in the sklearn package ([3]). It is an impurity-based method. There, the feature importance is explained as a reduction of impurity by use of the feature weighted by the probability of reaching the respective nodes. It is calculated as the average of the normalized importance values over all decision trees. In a decision tree, feature importance f of a feature i is calculated as in eq. 8.9.

$$f_i = \frac{\sum_{j:\text{node } j \text{ splits on feature } i} n_j}{\sum_{j \in \text{nodes}} n_j}, \tag{8.9}$$

where n_j is importance of node j which is calculated as in eq. 8.10.

$$n_j = w_j I_G(j) - w_{j_l} I_G(j_l) - w_{j_r} I_G(j_r), \tag{8.10}$$

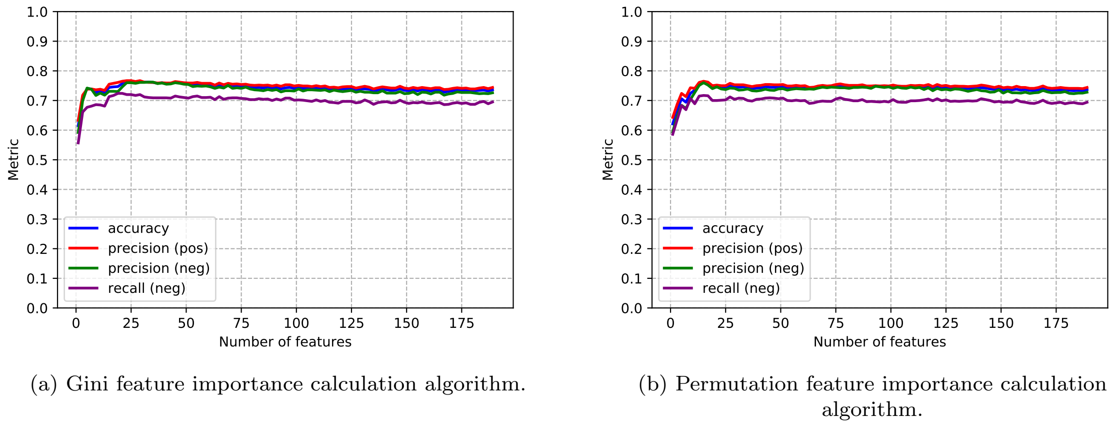
where $I_G(j)$ is Gini impurity value for the node j, w_j — weighted number of samples reaching the node j and j_l and j_r are the left and right children of the node j respectively.

Thus, we have discussed 3 algorithms for calculation of feature importance in random forests.

Choice of the algorithm for feature importance calculation

After describing the possible feature selection methods, we have to choose the one which we would like to follow for feature selection.

For a random forest classifier, the Shapley feature importance method returns similar results to the Gini feature importance algorithm. Therefore, let's investigate feature selection based only on the Gini method and the permutation method.



(a) Gini feature importance calculation algorithm. (b) Permutation feature importance calculation algorithm.

Figure 8.5: Evaluation of the models with threshold 0.5 with different number of features selected sequentially according to respective feature importance calculation method.

We will conduct the feature selection process in the following manner. Firstly, we calculate feature importance values according to both methods in the models with 189 uncorrelated features discussed in the subsection about correlated features in 8.2.3. We perform 10 simulations and average the feature importance values over these simulations. Then, we sort the features in descending order according to these feature importance values. Then, we include features one by one into the model according to this order and evaluate the resulting model. We use the random forest classification model with 2 classes and with parameters $(200, 15, 3, 2, \text{auto})$. Depending on the threshold, a different number of features appears to be optimal. Let's first consider the threshold 0.5 which maximizes accuracy. In Fig. 8.5 the metrics are presented for various numbers of included features selected according to the feature importance values calculated using 2 different methods.

We can see that quite fast, somewhere after including the 10 most important features, the model accuracy, both precision values, and negative recall stop growing rapidly. Then, we occasionally obtain high values of these metrics until the inclusion of 30-40 features. After that, all of them stagnate or even decrease slightly due to possible overfitting on features. If we follow the order obtained using the Gini index, we reach high values of accuracy around 0.76 already after the inclusion of 25 features and these numbers fall to 0.75 if we continue the process. On the other hand, when we use the order according to the permutation feature importance values, we can see that the accuracy grows slightly faster and reaches the value of 0.76 after the inclusion of around 15 features. Afterwards, it drops slightly and stays in the region of 0.75. We can see that slightly fewer features are required to reach the maximum accuracy and precision values when following the order obtained using the permutation feature importance calculation algorithm. However, the maximum values are similar in both methods, so, it is only a question of computational complexity: the smaller number of features, the faster is the algorithm. Thus, it is more profitable to use the permutation feature importance order in the case of the threshold 0.5.

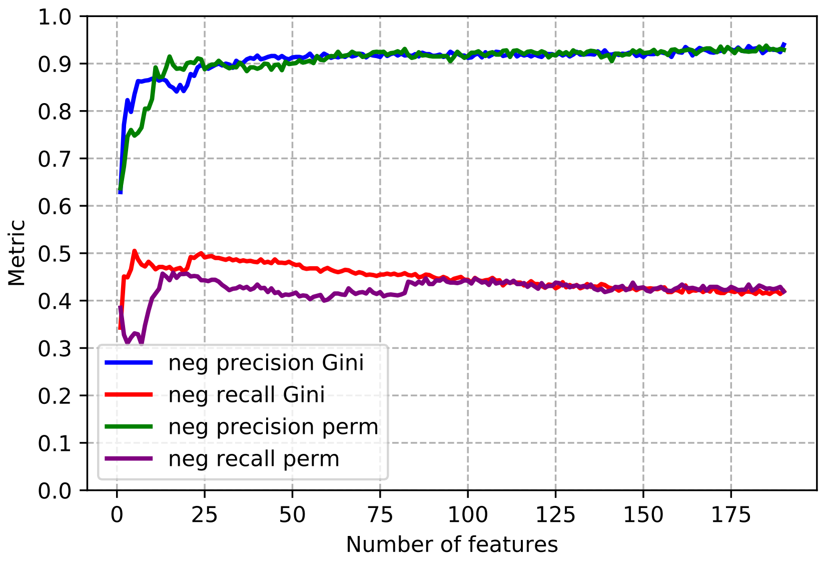


Figure 8.6: Evaluation of the model with the threshold 0.3 with different number of features selected sequentially according to respective feature importance calculation method.

However, in this project, we are more interested in the smaller thresholds which increase negative precision and recall. For example, let's look at the threshold 0.3 (Fig. 8.6). Until the inclusion of around 80 features, the selection according to Gini feature importance order results in a higher negative recall. It seems that the features on the positions 20-24 in the Gini feature importance order appear to be more important than predicted by the feature importance algorithm. Thus, we can consider including them earlier if we are going to use less than 25 features. Otherwise, we can simply follow the Gini feature importance order which is performing better than the permutation feature importance order on all other numbers of features excluding 15-20.

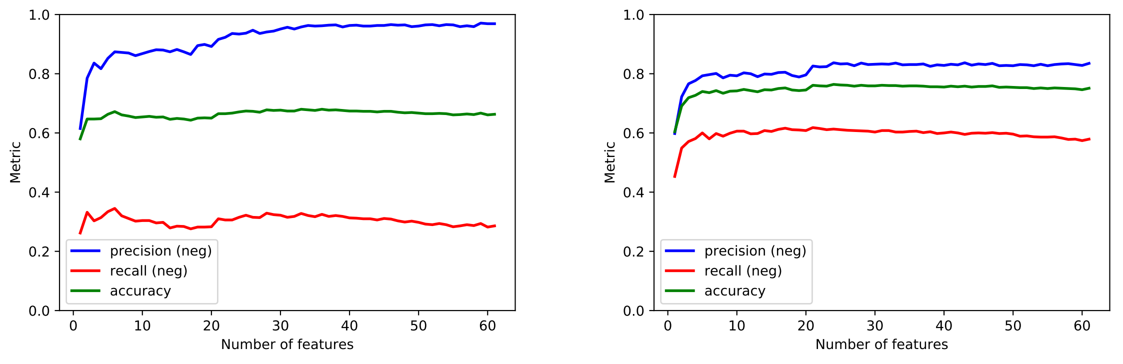
Furthermore, we can see that after the inclusion of 50 features the negative precision grows very slowly while the negative recall drops gradually. The negative precision reaches 0.92 with the standard deviation of 0.01 at that point and the highest negative precision when all the features are included is 0.93 ± 0.015. Therefore, we do not gain much in terms of the precision while losing a lot in terms of the recall if we continue including features into the classification model. Thus, we would consider not more than 50 features in further research.

In this section, we have compared different feature importance calculation algorithms. We have concluded that for the threshold of 0.5 it is more profitable to follow the permutation feature importance order. However, for the smaller thresholds such as, for example, 0.3, it is better to follow the Gini feature importance order. As we are interested in the smaller thresholds, we decide to follow the Gini feature importance order. Besides this, we have decided to investigate only the models with not more than 50 features.

Feature importance calculation results

After choosing the feature importance order which we are following, we look for the optimal number of features for different thresholds. As we have decided in the previous subsection, we consider not more than the first 50 features for classification.

Based on Fig. 8.6 we can choose to use first 50 features for the threshold of 0.3. We can also define the reasonable amount of features for other thresholds: 0.2 (Fig. 8.7a) and 0.4 (Fig. 8.7b). The results are presented in Table 8.2 where we refer to the negative precision and recall simply as precision and recall. All the numbers are the mean values with the standard deviation of approximately 0.01. Here, we can see that for all the thresholds we gain in terms of recall and almost do not lose anything in terms of precision when we keep the suggested reasonable number of features in the model. Thus, we retain the similar quality of the models in terms of precision while significantly increasing the quality in terms of the recall.



(a) Metric values for the models with the threshold 0.2. (b) Metric values for the models with the threshold 0.4.

Figure 8.7: Evaluation of the models with different number of features selected sequentially according to the Gini feature importance with the thresholds 0.2 and 0.4 for classification.

Threshold	Reasonable choice			Highest precision			All features	
	Features	Precision	Recall	Features	Precision	Recall	Precision	Recall
0.2	38	0.965	0.32	100	0.986	0.25	0.97	0.22
0.3	50	0.92	0.48	180	0.937	0.42	0.93	0.415
0.4	25	0.837	0.613	100	0.843	0.59	0.824	0.566

Table 8.2: Comparison of the means of negative precision and recall over 20 simulations for the models with different thresholds and number of features.

Thus, depending on the threshold that we would like to use, we could include different numbers of features in the models. The decision about the threshold should be made taking into account both the negative precision and the negative recall. This poses a typical precision-recall problem when we have to decide what is more important in the problem: having high recall or having high precision. Notice that here we recognize about 60% of the negative examples with a threshold 0.4, almost 50% with a threshold 0.3, and around 30% with a threshold 0.2. However, the threshold 0.2 results in higher precision values.

Thus, for the optimal performance of the models with thresholds 0.2, 0.3 or 0.4 we need not more than 50 features. Hence, anyway, we should omit at least 139 features which do not add much value to the classification model.

Currently, as the 50 most important features according to the Gini feature importance there are:

- S3 spatial sharpness and its 5 histogram features;
- 3 Laplace sharpness regional histogram features;
- S-index;
- Width of the image;
- 10 simple IL-NIQE Gabor features and 3 horizontal gradient IL-NIQE Gabor features;
- 4 BRISQUE features;
- 4 horizontal gradient IL-NIQE features;
- 2 IL-NIQE color features;
- 6 font thickness features;
- 2 border distance features calculated using morphological method;
- Brightness feature;
- Contrast feature;
- Hue feature;
- 2 PP, 1 Sauvola, 1 morphology and 1 CV orientation features;

Thus, we have come up with a reasonable number of features for the classification models with different thresholds. In addition to improving the recall of the models, reducing the number of features also reduces the computational complexity of the classification algorithm as we have to extract fewer features.

Computational complexity of classification algorithm

In this subsection, we look at the computational complexity of the classification algorithm with different numbers of selected features.

The random forest classification algorithm itself takes approximately 0.12 seconds. Thus, the features extraction part is the most computationally expensive. We remember from Table 7.2 that there were many computationally expensive features such as IL-NIQE features and s-index. When we include the features in the model according to the Gini feature importance order, we get a computationally expensive model for an image of a typical size 2500×3200 already after including a small number of features (Fig. 8.8). We can see that most of the features in the beginning are computationally expensive while for the less important features we spend less time on calculations. This is happening as many feature extraction algorithms extract several features simultaneously and one of these features can be important and others less important. For the latter ones, we do not have to repeat the calculations: we already have extracted these features.

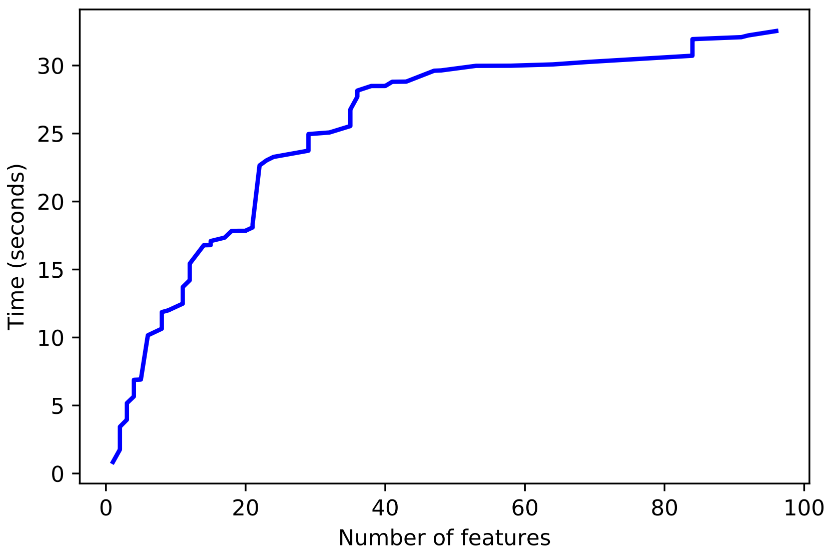


Figure 8.8: Computational complexity (in seconds) for the calculation of different number of features taken in the order of the Gini feature importance for an image of size 2500×3200.

We can conclude that the extraction of the features from the original images is a very computationally expensive process. From Figure 8.8 we can see that the extraction of 10 features already results in a model that takes around 10 seconds to run. As we are looking for a fast algorithm with calculations time around 1 second, the current classification algorithm is not a good solution. As one of the methods to reduce computational complexity, we can consider extracting the features from downscaled images.

8.2.4 Evaluation of the models with features extracted from downscaled images

As we have seen in the previous subsection about computational complexity, the feature extraction process is unaffordably slow. One of the reasons is that we use the original size of the images for feature extraction. If we downscale the images before extracting features, it will speed up the process. Thus, we are interested in the impact of using downscaled images for feature extraction on the performance of the models.

Let's evaluate the models where for extraction of some features we use downscaled images. We will refer to the features extracted from downscaled images as to "downscaled features". We will consider image sizes 1200×900 ("ds1"), 800×600 ("ds2"), and 600×500 ("ds3"). Let's also fix the parameters $(400, 15, 3, 2, \text{auto})$ of the random forest model and explore thresholds 0.2, 0.3, and 0.4. Depending on the threshold, we choose the number of features according to the reasonable number of features from Table 8.2. We extract the features with the highest computational complexity (IL-NIQE features, BRISQUE features, S-index, S3 sharpness, Laplace sharpness, orientation features, borders features, contrast, and brightness) from the downscaled images and compare the accuracy, negative precision, and recall of the obtained models. Besides this, we try to extract all the necessary features from downscaled images and consider this model, too. The complete results are presented in the Appendix A. Besides the metrics, we mention the gain in time when using respective downscaled features. There, we color with orange the cells with the evaluation metrics values that are significantly lower than the values for the original model (with difference larger than one standard deviation). That means, that downscaling the respective feature to the specified sizes results in deterioration of the model.

We can see that for each threshold the precision, recall, and accuracy metrics decrease significantly when we use all downscaled features compared to original features. That is why we investigated the models with only specific downscaled features.

We also notice that for most of the features the evaluation metrics do not depend on the size of the image from which the feature is extracted. The slight differences are mostly random as, for example, the lowest value of precision is sometimes obtained on the size "ds2" and not on "ds3" (as in Gradient Gabor in Tab. A.3). However, there are several exceptions here: gradient IL-NIQE features and BRISQUE features for the model with threshold 0.2, simple Gabor features for the model with threshold 0.3, and S3 sharpness and BRISQUE features in the model with threshold 0.4. For these features, we can observe a slight decrease in the metrics values with the image size. The decrease of the accuracy and negative precision with the size of the images is also observed in the models with all downscaled features.

Thus, among the suggested sizes, for some of the features, it is more profitable to consider the size "ds3" as it does not decrease the performance. For the features where the performance deteriorates with the size, we can choose different sizes of the images in order to balance the computational complexity and performance. However, for some very computationally expensive features, the loss in performance is unavoidable. Before making the final decision on which downscaled features to use, we evaluate the respective model again with the chosen downscaled features.

From the tables in Appendix A, we can see that it is possible to decrease the computational complexity of all orientation features, Laplace sharpness features, brightness, and contrast in every model without loss of quality of the model. Thus, we will use “ds3” size of the images for the extraction of these features. For other features, it depends on the model if the quality decreases when using downscaled features.

For each model we choose if to use the downscaled feature instead of the original according to the following rules. We choose “ds3” size of the image if all of the metric values are not significantly smaller compared to the original model. If the decrease for some image sizes is significant for some metric, we choose size “ds1”, “ds2” or the original feature depending on the computational complexity. For example, for S3 regional sharpness it is better to use original size as otherwise we obtain a significant loss in precision for the models with thresholds 0.2 and 0.3. On the other hand, for S-index in the model with the threshold 0.2 we can use “ds1” size. Moreover, we should consider that simple and gradient Gabor features can’t be calculated for different image sizes: this will increase the computational complexity as we will have to construct and apply log-Gabor filters of different sizes.

Therefore, we choose to use downscaled to “ds3” orientation features, brightness, contrast, and Laplace regional sharpness features for all thresholds. For other features the choice of the downscaled features depends on the threshold. We use the following downscaled features in different models:

- With the threshold 0.2:
  - “ds3” of S3 sharpness, morphological borders, and IL-NIQE color features;
  - “ds1” of BRISQUE, S-index, simple and gradient Gabor features, and gradient IL-NIQE features;
- With the threshold 0.3:
  - “ds3” of S3 sharpness, morphological borders, S-index, BRISQUE, IL-NIQE color features, and gradient IL-NIQE features;
  - “ds1” of simple Gabor and gradient Gabor features;
- With the threshold 0.4:
  - “ds3” of regional S3 sharpness, S-index, simple Gabor and gradient Gabor features, IL-NIQE color features, and gradient IL-NIQE features;
  - “ds1” of S3 sharpness, morphological borders, and BRISQUE;

According to this choice, we can evaluate the respective suggested models. The results can be found in Table 8.3. We observe the increase in the speed of the classification algorithm, though the performance deteriorates sometimes. For the thresholds of 0.2 and 0.3 we do not lose in the negative precision but lose in the negative recall. For the model with the threshold of 0.4, we lose both in the negative precision and recall. On the other hand, we can save up to 26 seconds using downscaled features.

However, we can still see that for the models with the thresholds 0.2 and 0.3 the calculation time is still relatively long. In order to decrease it, we have to use downscaled to at least “ds2”

Model	Threshold	Accuracy		Precision neg.		Recall neg.		Time
		mean	std	mean	std	mean	std	(sec)
Original	0.2	0.675	0.004	0.968	0.008	0.312	0.009	28.616
Suggested model	0.2	0.661	0.004	0.965	0.005	0.283	0.01	3.877
Incl. ds2 for IL-NIQE	0.2	0.66	0.004	0.962	0.01	0.281	0.008	2.165
Incl. 400 × 300 features	0.2	0.658	0.005	0.96	0.011	0.276	0.01	2.069
All downscaled features	0.2	0.63	0.003	0.912	0.012	0.228	0.007	1.173
Original	0.3	0.737	0.006	0.919	0.009	0.476	0.01	29.761
Suggested model	0.3	0.709	0.005	0.92	0.007	0.412	0.01	3.039
Incl. ds2 for IL-NIQE	0.3	0.713	0.004	0.92	0.004	0.42	0.009	1.975
Incl. 400 × 300 features	0.3	0.714	0.005	0.92	0.006	0.423	0.011	1.799
All downscaled features	0.3	0.686	0.003	0.86	0.005	0.39	0.007	1.198
Original	0.4	0.762	0.003	0.841	0.006	0.602	0.005	23.402
Suggested model	0.4	0.739	0.004	0.819	0.006	0.564	0.004	1.417
Incl. 400 × 300 features	0.4	0.738	0.005	0.818	0.008	0.563	0.008	1.240
All downscaled features	0.4	0.717	0.002	0.774	0.006	0.554	0.004	0.986

Table 8.3: Evaluation of the suggested models based on the metric values obtained using downscaled features and comparison to the original models and the models with all downscaled features.

size IL-NIQE features and to “ds3” s-index as these are the most computationally expensive. Otherwise, we will not reduce computational complexity enough. From the same Table 8.3 we can see that this does not influence the performance much but reduces the calculations time (rows with “incl. ds2 for IL-NIQE”).

Even more improvement in terms of computational complexity can be obtained by using an even smaller size of downscaled images for the extraction of the computationally complex features. Thus, let’s use images of size 400 × 300 for extraction of the most computationally expensive features at the moment. These are s-index and gradient IL-NIQE features. However, gradient IL-NIQE features are extracted from 400 × 300 image only for the thresholds 0.3 and 0.4 as for 0.2 extracting them from small images deteriorates performance (Tab. A.3). The comparison of these models is also given in Table 8.3 in the rows “Incl. 400 × 300 features”. Again, as in the previous case, we do not lose much in the quality while saving additionally approximately 0.1 second in each model. This process can be continued for less computationally expensive features for speeding up the algorithm. However, we will use the current ones as the further speeding up requires more experiments and will not save that much time.

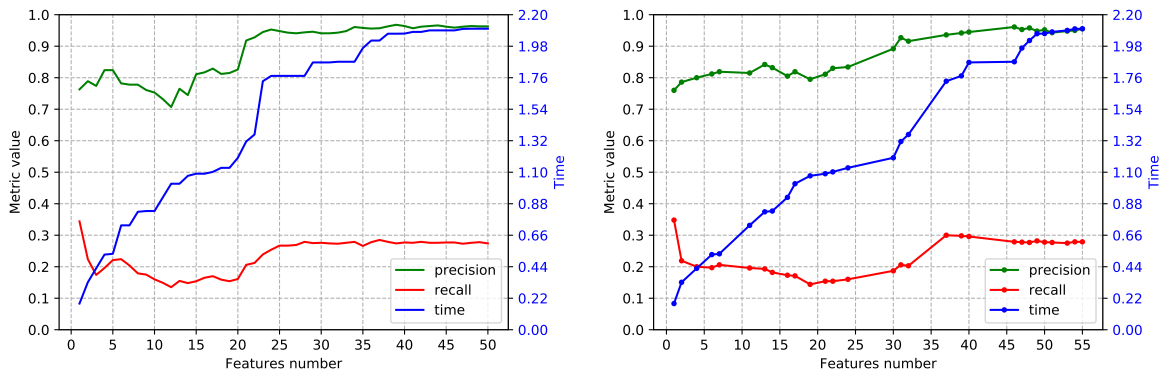
We can also see that the final suggested models provide better quality than the models with all downscaled features and have only slightly larger computational complexity.

Thus, in this section, we have suggested using different sizes of images for the extraction of different features. These sizes differ for the models with different thresholds. This slightly worsens the quality of the classification algorithms but improves a lot on their computational complexity.

8.2.5 Changing the order of feature inclusion

In Section 8.2.4 we have suggested the models with the reduced computational complexity. However, the calculations time of 2 seconds in some models is still too long. That is why we have to consider other options to speed up the classification algorithm. One of the options is to change the order of inclusion of the features in the models.

In Section 8.2.3 we selected features one by one according to the Gini feature importance index. However, as we mentioned in that section, some of the features are extracted simultaneously: for example, histogram features of regional sharpness or pairs of the parameters from IL-NIQE Gabor features. Some of these features are more important than others and, hence, included in the model at different moments. The suggestion is to include these groups of features into the model together at the moment when the first feature of the group is included according to the feature importance order. This method can have 2 variations. In the first one, we include all the features from the group, for example, all 9 S3 regional sharpness features. In the second variation, we include only the features which appear in the 50 most important features. The experiments showed that these 2 methods result in the same performance of the model. Thus, we will use the latter variation as it involves less features. We call this a model with grouped feature selection order.

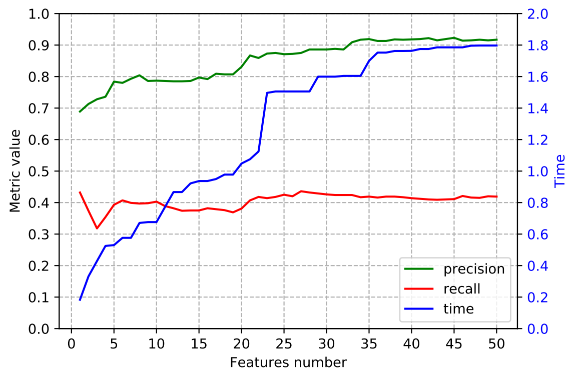


(a) Model with initial features selection order. (b) Model with grouped feature selection order.

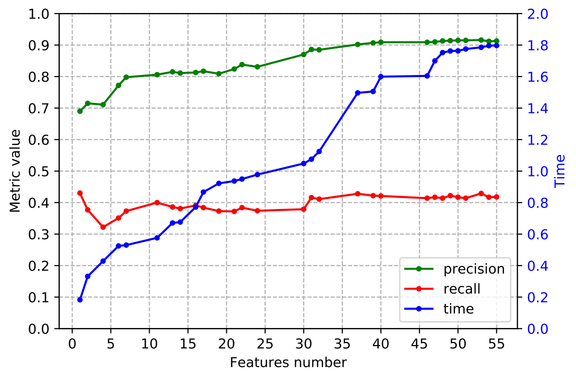
Figure 8.9: Metric values and calculation time of the models with the threshold 0.2.

For each of the thresholds 0.2, 0.3, and 0.4 let's compare the initially suggested model and the model with grouped feature selection order. The plots for evaluation metrics and computational complexity can be seen in Figures 8.9-8.11. In the models with grouped feature selection order, the dots correspond to the moments when a group of features is added. Thus, we cannot stop the selection process somewhere between the dots and can do it only at the moments corresponding to the dots. From these graphs, we can select the optimal number of features based on the negative precision, negative recall, and computational complexity. As we aim at the calculations time close to 1 second, we choose the number of features according to it.

In the models with the threshold 0.2 (Fig. 8.9) we stop the selection on 22 features for initial features ordering and on 32 features for the grouped ordering which corresponds to the same calculations time of 1.355 seconds. If we stop earlier when the calculations time is 1.1, we lose in the negative precision and negative recall around 0.1 which is quite a large loss. Thus, it is decided to spend additional 0.2 seconds. At that moment, there is no significant difference in negative precision and recall in the 2 models with different feature inclusion orders as can be seen from Table 8.4. We can also notice that the negative precision and recall grow after the inclusion of the computationally expensive regional S3 sharpness features up to 0.936 and 0.3 respectively. Afterwards, the negative recall stays stable and the negative precision grows up to 0.96.

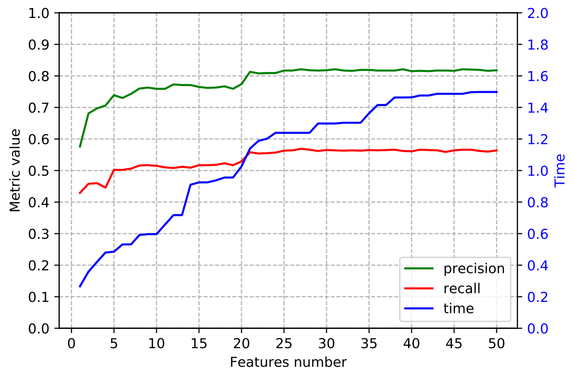


(a) Model with initial features selection order.

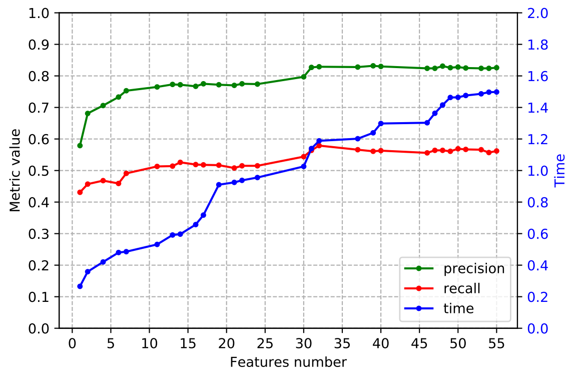


(b) Model with grouped feature selection order.

Figure 8.10: Metric values and calculation time of the models with the threshold 0.3.



(a) Model with initial features selection order.



(b) Model with grouped feature selection order.

Figure 8.11: Metric values and calculation time of the models with the threshold 0.4.

We can avoid the expensive computations of S3 regional sharpness by use of downscaled S3 regional sharpness features or by removing them completely and including the next features. Both of the methods give the same outcome in terms of the quality, so, we choose a faster method, where we remove these features completely. Let's include the following, less computationally expensive, feature instead of S3 regional sharpness. Then, we save around 0.3 seconds and we observe the same gain in the negative precision and a lower gain in the negative recall compared to what we get in the model with included S3 regional sharpness. Moreover, the evaluation metrics values do not grow further if we include more features and we can't reach 0.96 precision as it was observed with S3 regional sharpness features. Therefore, the optimal number of the features in the model with the threshold 0.2 is 34 with following the grouped feature selection order without S3 regional sharpness features.

We perform the same comparisons for the models with the thresholds 0.3 and 0.4. The main difference in these models, compared to the one with the threshold 0.2 is that we observe a difference in the performance of the models with initial and grouped feature selection order. Spending the same time on the features calculations, we get higher negative precision and recall for the models with grouped feature selection order (Tab. 8.4 and Figs. 8.10-8.11).

In the model with the threshold 0.3, we also stop after the inclusion of 22 features in the initial order and 32 in the grouped feature order. Then, we perform the same trick with the exclusion of S3 regional sharpness and inclusion of the next features from the grouped feature selection order. Similarly to the case with the threshold 0.2, this gives an increase in the negative precision and recall up to 0.905 and 0.426 respectively. These metrics values do not increase if we continue including other features as the S3 regional sharpness features are absent. Thus, here, the optimal choice of the number of features is the same: 34 features taken in the grouped feature selection order without S3 regional sharpness.

In the model with the threshold 0.4, we do not observe any growth of the evaluation metrics after the moment when we include 32 features according to the grouped feature selection order. Thus, there is no need to further extend the feature set in this case and the optimal number of features is 32 taken in the grouped feature selection order.

Model	Threshold	# features	Precision neg.		Recall neg.		Time
			mean	std	mean	std	(sec)
Initial	0.2	22	0.928	0.018	0.212	0.008	1.355
Grouped	0.2	32	0.916	0.015	0.203	0.007	1.355
Grouped with reg. S3	0.2	37	0.936	0.013	0.3	0.011	1.727
Grouped no reg. S3	0.2	34	0.938	0.011	0.249	0.01	1.392
Initial	0.3	22	0.859	0.007	0.418	0.007	1.126
Grouped	0.3	32	0.885	0.009	0.411	0.008	1.126
Grouped with reg. S3	0.3	37	0.902	0.006	0.428	0.009	1.498
Grouped no reg. S3	0.3	34	0.905	0.007	0.426	0.007	1.135
Initial	0.4	22	0.808	0.007	0.554	0.007	1.190
Grouped	0.4	32	0.829	0.007	0.579	0.009	1.190

Table 8.4: Comparison of the models with initial feature selection order and with grouped feature selection order where we use different number of features.

Thus, the 32 features, which are used in these models, are the following (in the grouped feature selection order):

1. S3 sharpness;
2. Shape ν gradient IL-NIQE Gabor feature with scale 1/8 and orientation $3\pi/5$;
3. Shape ν and variance σ^2 gradient IL-NIQE Gabor features with scale 1/2 and orientation 0;
4. Shape ν and variance σ^2 simple IL-NIQE Gabor features with scale 1/4 and orientation $\pi/5$;

5. Maximum brightness;
6. Shape ν and variance σ^2 gradient IL-NIQE features for channels O_1 and O_2;
7. Shape ν and variance σ^2 simple IL-NIQE Gabor features with scale 1/8 and orientation 0;
8. Contrast ratio feature;
9. Shape ν and variance σ^2 simple IL-NIQE Gabor features with scale 1/2 and orientation $2\pi/5$;
10. Variance σ^2 gradient IL-NIQE Gabor feature with scale 1/32 and orientation 0;
11. Distance from the text to the right border, in the whole image and in the middle of the image, using morphological method;
12. Projection profile skew histograms at $2°$ and $> 3°$;
13. Complexity variance skew histogram at $0°$;
14. Shape ν and variance σ^2 simple IL-NIQE Gabor features with scale 1/2 and orientation 0;
15. Font mean thickness;
16. Font thickness mean histogram value at $[1; 1.5]$;
17. Font thickness std histogram values at $[0; 0.5], [0.5; 1], [2; 2.5], [3; 3.5]$;
18. BRISQUE shape ν for mscn coefficients;
19. S-index;

The 33-34 features used in two out of 3 models are BRISQUE shape ν and left variance σ^2_{left} for vertical gradient image.

Here, font features are extracted from the original image. Brightness, contrast, PP skew features, and CV skew feature are extracted from the downscaled image of size 600×500. S-index is extracted from the downscaled image of size 400×300. Other features are extracted from different downscaled images depending on the threshold in the model. More specifically:

- The model with the threshold 0.2:
  - Image of size 1200×900 for BRISQUE features;
  - Image of size 800×600 for simple and gradient IL-NIQE Gabor and gradient IL-NIQE features;
  - Image of size 600×500 for S3 sharpness, morphological borders, and IL-NIQE color features;
- The model with the threshold 0.3:
  - Image of size 800×600 for simple and gradient IL-NIQE Gabor features;

  - Image of size 600 × 500 for S3 sharpness, morphological borders, BRISQUE, and IL-NIQE color features;
  - Image of size 400 × 300 for gradient IL-NIQE features;

- The model with the threshold 0.4:
  - Image of size 1200 × 900 for S3 sharpness, morphological borders, and BRISQUE features;
  - Image of size 600 × 500 for regional S3 sharpness, simple and gradient IL-NIQE Gabor features, and IL-NIQE color features;
  - Image of size 400 × 300 for gradient IL-NIQE features;

To sum up, we have developed the models for the classification of the images into 2 classes using random forest classification algorithms with different probability threshold values. We have performed the feature selection process and suggested improvements of the algorithm for speeding it up only with a slight loss in performance.

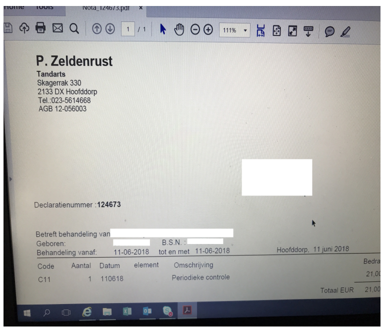
8.2.6 Incorrectly classified images analysis

In this section, we will look into examples of incorrectly classified images, explain probable reasons for being incorrectly classified, and discuss possible new features that might help to classify these examples correctly in the future. We will use classification into 2 classes with parameters $(400, 15, 3, 2, \text{auto})$. As we will look into the probabilities of being in the positive class, the thresholds do not play any role in this section. We will consider 6 incorrectly classified images for each class with the highest probability of being in another class. That is to say, 6 images from the positive class with the highest predicted probability of being negative and 6 images from the negative class with the highest predicted probability of being positive.

To begin with, let's look into the examples from the positive class that are predicted to be negative with the highest probabilities. They are presented in Figure 8.12 with predicted probability p of being included in the positive class (meaning that $1 - p$ is the predicted probability of them being included in the negative class). For some of the examples incorrect classification can be explained. For example, the image in Fig. 8.12a is a photo of the screen. Photos of a screen are negative examples in the vast majority of cases. Thus, this image is an exception. The image in Fig. 8.12f is not a photo of an invoice but of a check and, thus, it is correctly rejected by the ORTEC algorithm. However, it is classified as a negative example as it does not look similar to invoices: for example, does not have enough white background and has a wide range of colors. Probably, if we will have enough correctly rejected examples of this type, we will classify them as positive. For the remaining examples brightness, sharpness, and some IL-NIQE features are the most influential in the decision of classifying them into the negative class.

We can also notice that there is only one image from the positive class with the predicted probabilities p of belonging to the negative class lower than 0.2 and only 4 images with $p < 0.3$. That means, that there are few false positives in the models with the thresholds 0.2 and 0.3.

Next, let's also look into the examples of images from the negative class that are incorrectly classified as positive with the highest probabilities. They are presented in Figure 8.13. Here,


Nota_124673.pdf

P. Zeldenrust
Tandarts
Skagerrak 330
2133 DX Hoofddorp
Tel.:023-5614668
AGB 12-056003

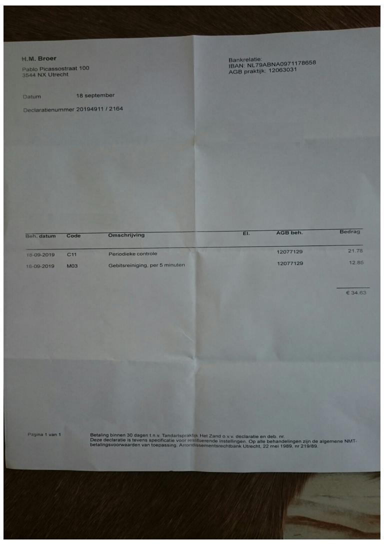
[redacted]

Declaratienummer :**124673**

Betreft behandeling van [redacted]
Geboren: [redacted] B.S.N. : [redacted]
Behandeling vanaf: 11-06-2018 tot en met 11-06-2018 Hoofddorp, 11 juni 2018

Code	Aantal	Datum	element	Omschrijving	Bedra
C11	1	110618		Periodieke controle	21,0
				Totaal EUR	21,0

(a) $p = 0.184$


H.M. Broer
Pablo Picassostraat 100
3544 NX Utrecht

Bankrelatie:
IBAN: NL79ABNA0971178658
AGB praktijk: 12063031

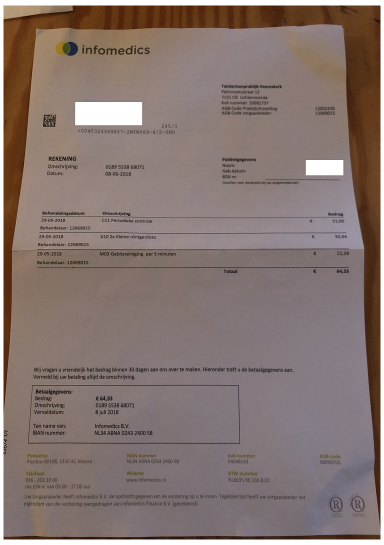
Datum 18 september
Declaratienummer 20194911 / 2164

Beh. datum	Code	Omschrijving	El.	AGB beh.	Bedrag
18-09-2019	C11	Periodieke controle		12077129	21.78
18-09-2019	M03	Gebitsreiniging, per 5 minuten		12077129	12.85
					€ 34.63

Pagina 1 van 1

Betaling binnen 30 dagen t.n.v. Tandartspraktijk Het Zand o.v.v. declaratie en deb. nr.
Deze declaratie is tevens specificatie voor restituerende instellingen. Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing. Arrondissementsrechtbank Utrecht, 22 mei 1989, nr 219/89.

(b) $p = 0.234$


infomedics

Tandartsenpraktijk Heemskerk
Patronaatsstraat 12
7131 CG Lichtenvoorde
KvK-nummer: 50681737
AGB-Code Praktijk/Instelling: 12002326
AGB-Code zorgaanbieder: 12069015

[redacted]
245/3
<0045326969857-2WTW069-4/2-SND

REKENING
Omschrijving: 0189 5538 68071
Datum: 08-06-2018

Patiëntgegevens
Naam: [redacted]
Geb.datum:
BSN nr:
(invullen voor declaratie bij uw zorgverzekeraar)

Behandelingsdatum	Omschrijving		Bedrag
29-05-2018 Behandelaar: 12069015	C11 Periodieke controle	€	21,00
29-05-2018 Behandelaar: 12069015	X10 2x Kleine röntgenfoto	€	30,94
29-05-2018 Behandelaar: 12069015	M03 Gebitsreiniging, per 5 minuten	€	12,39
	Totaal	€	64,33

Wij vragen u vriendelijk het bedrag binnen 30 dagen aan ons over te maken. Hieronder treft u de betaalgegevens aan. Vermeld bij uw betaling altijd de omschrijving.

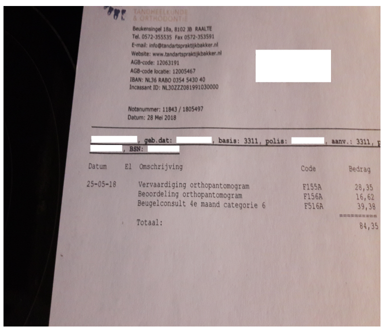
Betaalgegevens:	
Bedrag:	**€ 64,33**
Omschrijving:	0189 5538 68071
Vervaldatum:	8 juli 2018
Ten name van:	Infomedics B.V.
IBAN nummer:	NL34 ABNA 0243 2400 58

63574 1/1

Postadres	IBAN nummer	KvK-nummer	AGB-code
Postbus 60108, 1320 AC Almere	NL34 ABNA 0243 2400 58	04048143	98098705
Telefoon	Website	BTW-nummer	
036 - 203 19 00 ma t/m vr van 09.00 - 17.00 uur	www.infomedics.nl	NL8035.98.130.B.01	

Uw zorgaanbieder heeft Infomedics B.V. de opdracht gegeven om de vordering op u te innen. Tegelijkertijd heeft uw zorgaanbieder het eigendom van die vordering overgedragen aan Infomedics Finance B.V. (gecedeerd).

(c) $p = 0.259$


TANDHEELKUNDE & ORTHODONTIE
Beukensingel 18a, 8102 JB RAALTE
Tel. 0572-355535 Fax 0572-353591
E-mail: info@tandartspraktijkbakker.nl
Website: www.tandartspraktijkbakker.nl
AGB-code: 12063191
AGB-code locatie: 12005467
IBAN: NL36 RABO 0354 5430 40
Incassant ID: NL30ZZZ081991030000

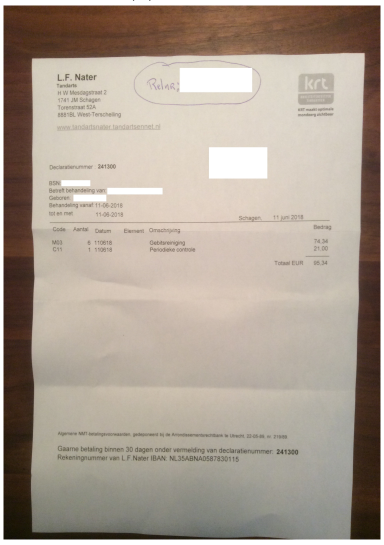
[redacted]

Notanummer: 11843 / 1805497
Datum: 28 Mei 2018

[redacted], geb.dat: [redacted], basis: 3311, polis: [redacted], aanv.: 3311,
[redacted], BSN: [redacted]

```
Datum     El  Omschrijving                              Code      Bedrag

25-05-18      Vervaardiging orthopantomogram            F155A      28,35
              Beoordeling orthopantomogram              F156A      16,62
              Beugelconsult 4e maand categorie 6        F516A      39,38
                                                                ==========
              Totaal:                                              84,35
```

(d) $p = 0.296$


L.F. Nater
Tandarts
H W Mesdagstraat 2
1741 JM Schagen
Torenstraat 52A
8881BL West-Terschelling
www.tandartsnater.tandartsennet.nl

Relnr: [redacted]

krt
KRT maakt optimale mondzorg zichtbaar

[redacted]

Declaratienummer : **241300**

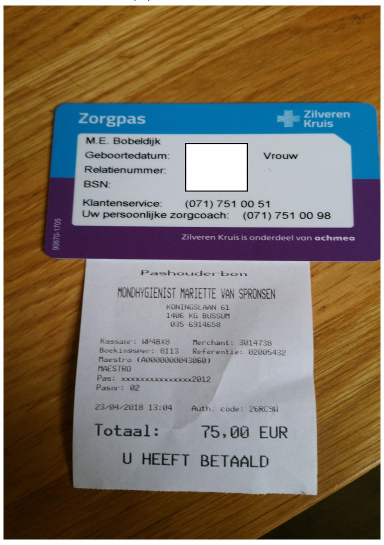
BSN: [redacted]
Betreft behandeling van: [redacted]
Geboren: [redacted]
Behandeling vanaf: 11-06-2018
tot en met 11-06-2018
Schagen, 11 juni 2018

Code	Aantal	Datum	Element	Omschrijving	Bedrag
M03	6	110618		Gebitsreiniging	74,34
C11	1	110618		Periodieke controle	21,00
				Totaal EUR	95,34

Algemene NMT-betalingsvoorwaarden, gedeponeerd bij de Arrondissementsrechtbank te Utrecht, 22-05-89, nr. 219/89

Gaarne betaling binnen 30 dagen onder vermelding van declaratienummer: **241300**
Rekeningnummer van L.F.Nater IBAN: NL35ABNA0587830115

(e) $p = 0.306$

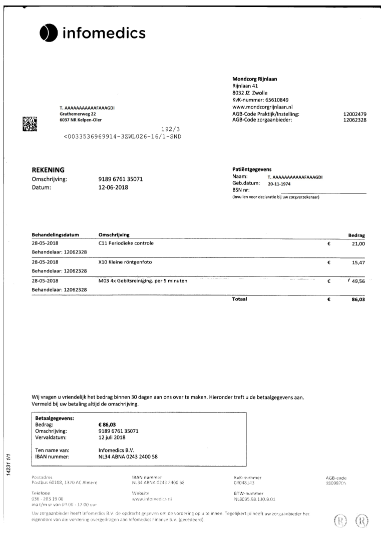

Zorgpas
Zilveren Kruis
M.E. Bobeldijk
Geboortedatum: [redacted] Vrouw
Relatienummer:
BSN:
Klantenservice: (071) 751 00 51
Uw persoonlijke zorgcoach: (071) 751 00 98
Zilveren Kruis is onderdeel van achmea
90870-1705

```
           Pashouderbon
 MONDHYGIENIST MARIETTE VAN SPRONSEN
          KONINGSLAAN 61
          1406 KG BUSSUM
           035-6914658

Kassanr: WP4BX8       Merchant: 3014738
Boekingsper: 8113     Referentie: 02005432
Maestro (A0000000043060)
MAESTRO
Pas: xxxxxxxxxxxxxxx2012
Pasnr: 02

23/04/2018 13:04      Auth. code: 26RCSW

Totaal:        75,00 EUR

     U HEEFT BETAALD
```

(f) $p = 0.312$

Figure 8.12: Examples of images incorrectly classified as negative.

we can also explain some of the problems. Firstly, in the image in Fig 8.13a there is a small tick before the cost in the third line. It is recognized as “1” and, hence, the total amount does not sum up. In Fig. 8.13b and Fig. 8.13e we have cut images that still cannot be correctly classified, probably due to the fact that we do not have enough cut examples in the dataset. In the images in Fig. 8.13c and Fig. 8.13f partially recognized images are presented. For the one in Fig. 8.13c some element number is missing. In Fig. 8.13f the format of the invoice is unusual and the treatment date cannot be found as it is expected to be in the line of the treatment and not above the treatment table. These problems are related to the content and layout of the invoice and not to the quality of the photo. Thus, they are not detected by our classification algorithm. Finally, in the image in Fig. 8.13d the anonymization box is not well aligned and covers the date of treatment in some lines so that the date cannot be read correctly by OCR.



infomedics

T. AAAAAAAAAAAFAAAGDI
Grathemerweg 22
6037 NR Kelpen-Oler

192/3
<0033536969914-3ZWL026-16/1-SND

Mondzorg Rijnlaan
Rijnlaan 41
8032 JZ Zwolle
KvK-nummer: 65610849
www.mondzorgrijnlaan.nl
AGB-Code Praktijk/Instelling: 12002479
AGB-Code zorgaanbieder: 12062328

REKENING
Omschrijving: 9189 6761 35071
Datum: 12-06-2018

Patiëntgegevens
Naam: T. AAAAAAAAAAAFAAAGDI
Geb.datum: 20-11-1974
BSN nr:
(invullen voor declaratie bij uw zorgverzekeraar)

Behandelingsdatum	Omschrijving		Bedrag
28-05-2018	C11 Periodieke controle	€	21,00
Behandelaar: 12062328			
28-05-2018	X10 Kleine röntgenfoto	€	15,47
Behandelaar: 12062328			
28-05-2018	M03 4x Gebitsreiniging. per 5 minuten	€	49,56
Behandelaar: 12062328			
	Totaal	€	86,03

Wij vragen u vriendelijk het bedrag binnen 30 dagen aan ons over te maken. Hieronder treft u de betaalgegevens aan. Vermeld bij uw betaling altijd de omschrijving.

Betaalgegevens:
Bedrag: € 86,03
Omschrijving: 9189 6761 35071
Vervaldatum: 12 juli 2018

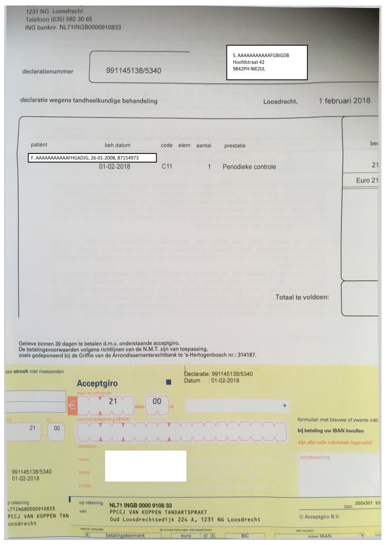
Ten name van: Infomedics B.V.
IBAN nummer: NL34 ABNA 0243 2400 58

14231 1/1

Postadres
Postbus 60108, 1320 AC Almere
IBAN nummer
NL34 ABNA 0243 2400 58
KvK-nummer
04048143
AGB-code
98098705
Telefoon
036 - 203 19 00
ma t/m vr van 09.00 - 17.00 uur
Website
www.infomedics.nl
BTW-nummer
NL8035.98.130.B.01

Uw zorgaanbieder heeft Infomedics B.V. de opdracht gegeven om de vordering op u te innen. Tegelijkertijd heeft uw zorgaanbieder het eigendom van die vordering overgedragen aan Infomedics Finance B.V. (gecedeerd).

(a) $p = 0.960$



1231 NG Loosdrecht
Telefoon (035) 582 30 65
ING banknr. NL71INGB0000910833

declaratienummer 991145138/5340

S. AAAAAAAAAAAFGBIGDB
Hoofdstraat 42
9842PH NIEZIJL

declaratie wegens tandheelkundige behandeling Loosdrecht, 1 februari 2018

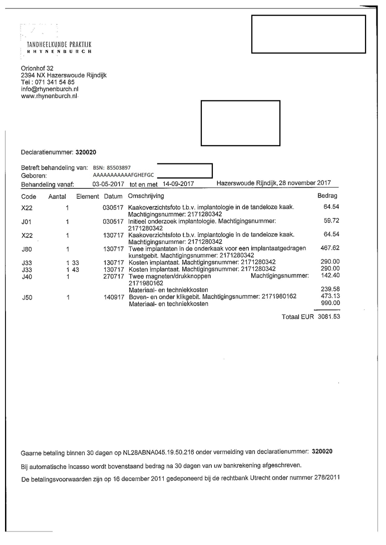
patiënt	beh.datum	code	elem.	aantal	prestatie	bed
F. AAAAAAAAAAAFHGADJG, 26-01-2008, 87154973						
	01-02-2018	C11		1	Periodieke controle	21
						Euro 21

Totaal te voldoen:

Gelieve binnen 30 dagen te betalen d.m.v. onderstaande acceptgiro.
De betalingsvoorwaarden volgens richtlijnen van de N.M.T. zijn van toepassing,
zoals gedeponeerd bij de Griffie van de Arrondissementsrechtbank te 's-Hertogenbosch nr.: 314187.

eze strook niet meezenden
Acceptgiro
Declaratie: 991145138/5340
Datum : 01-02-2018
over te schrijven
21 euro 00 ct
ro 21 ct 00
van bankrekening (IBAN)
formulier met blauwe of zwarte inkt
bij betaling uw IBAN invullen
zijn alle rode rubrieken ingevuld?
van/door
naam
adres
plaats
handtekening
991145138/5340
01-02-2018
p rekening
L71INGB0000910833
PCCJ VAN KOPPEN TAN
oosdrecht
op rekening NL71 INGB 0000 9108 33
van PPCCJ VAN KOPPEN TANDARTSPRAKT
Oud Loosdrechtsedijk 224 A, 1231 NG Loosdrecht
2004301 E0
D001
© Acceptgiro B.V.
nadruk verboden
de ruimte hieronder niet beschrijven
niet vouwen
X betalingskenmerk euro ct X BIC naar IBAN X

(b) $p = 0.911$



TANDHEELKUNDE PRAKTIJK
RHYNENBURCH

Orionhof 32
2394 NX Hazerswoude Rijndijk
Tel : 071 341 54 85
info@rhynenburch.nl
www.rhynenburch.nl

Declaratienummer: **320020**

Betreft behandeling van: BSN: 85503897
Geboren: AAAAAAAAAAAFGHEFGC
Behandeling vanaf: 03-05-2017 tot en met 14-09-2017 Hazerswoude Rijndijk, 28 november 2017

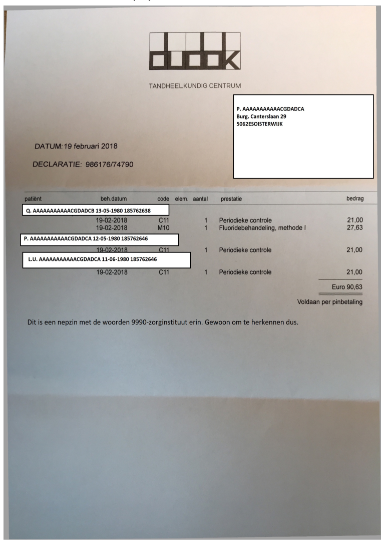
Code	Aantal	Element	Datum	Omschrijving	Bedrag
X22	1		030517	Kaakoverzichtsfoto t.b.v. implantologie in de tandeloze kaak. Machtigingsnummer: 2171280342	64.54
J01	1		030517	Initieel onderzoek implantologie. Machtigingsnummer: 2171280342	59.72
X22	1		130717	Kaakoverzichtsfoto t.b.v. implantologie in de tandeloze kaak. Machtigingsnummer: 2171280342	64.54
J80	1		130717	Twee implantaten in de onderkaak voor een implantaatgedragen kunstgebit. Machtigingsnummer: 2171280342	467.62
J33	1	33	130717	Kosten implantaat. Machtigingsnummer: 2171280342	290.00
J33	1	43	130717	Kosten implantaat. Machtigingsnummer: 2171280342	290.00
J40	1		270717	Twee magneten/drukknoppen Machtigingsnummer: 2171980162	142.40
				Materiaal- en techniekkosten	239.58
J50	1		140917	Boven- en onder klikgebit. Machtigingsnummer: 2171980162	473.13
				Materiaal- en techniekkosten	990.00
				Totaal EUR	3081.53

Gaarne betaling binnen 30 dagen op NL28ABNA045.19.50.216 onder vermelding van declaratienummer: **320020**

Bij automatische incasso wordt bovenstaand bedrag na 30 dagen van uw bankrekening afgeschreven.

De betalingsvoorwaarden zijn op 16 december 2011 gedeponeerd bij de rechtbank Utrecht onder nummer 278/2011

(c) $p = 0.9$



TANDHEELKUNDIG CENTRUM

P. AAAAAAAAAAACGDADCA
Burg. Canterslaan 29
5062ESOISTERWIJK

DATUM: 19 februari 2018

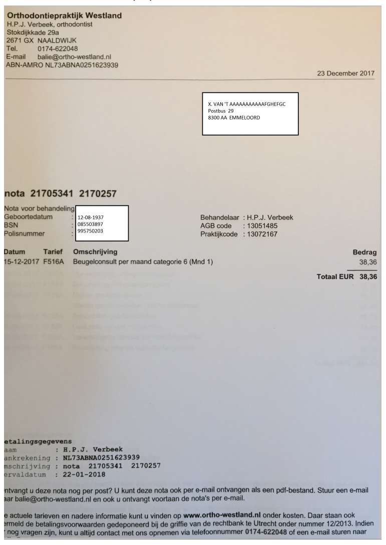
DECLARATIE: 986176/74790

patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
Q. AAAAAAAAAAACGDADCB 13-05-1980 185762638						
	19-02-2018	C11		1	Periodieke controle	21,00
	19-02-2018	M10		1	Fluoridebehandeling, methode I	27,63
P. AAAAAAAAAAACGDADCA 12-05-1980 185762646						
	19-02-2018	C11		1	Periodieke controle	21,00
L.U. AAAAAAAAAAACGDADCA 11-06-1980 185762646						
	19-02-2018	C11		1	Periodieke controle	21,00
						Euro 90,63

Voldaan per pinbetaling

Dit is een nepzin met de woorden 9990-zorginstituut erin. Gewoon om te herkennen dus.

(d) $p = 0.889$



Orthodontiepraktijk Westland
H.P.J. Verbeek, orthodontist
Stokdijkkade 29a
2671 GX NAALDWIJK
Tel. 0174-622048
E-mail balie@ortho-westland.nl
ABN-AMRO NL73ABNA0251623939

23 December 2017

X. VAN 'T AAAAAAAAAAAFGHEFGC
Postbus 29
8300 AA EMMELOORD

nota 21705341 2170257

Nota voor behandeling
Geboortedatum : 12-08-1937
BSN : 085503897
Polisnummer : 995750203

Behandelaar : H.P.J. Verbeek
AGB code : 13051485
Praktijkcode : 13072167

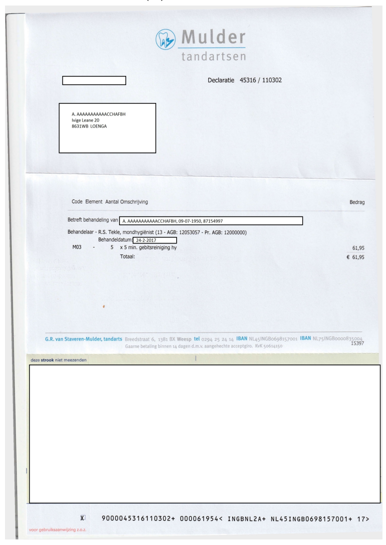
Datum	Tarief	Omschrijving	Bedrag
15-12-2017	F516A	Beugelconsult per maand categorie 6 (Mnd 1)	38,36
		Totaal EUR	**38,36**

etalingsgegevens
aam : **H.P.J. Verbeek**
ankrekening : **NL73ABNA0251623939**
mschrijving : **nota 21705341 2170257**
ervaldatum : **22-01-2018**

ntvangt u deze nota nog per post? U kunt deze nota ook per e-mail ontvangen als een pdf-bestand. Stuur een e-mail aar balie@ortho-westland.nl en ook u ontvangt voortaan de nota's per e-mail.

e actuele tarieven en nadere informatie kunt u vinden op **www.ortho-westland.nl** onder kosten. Daar staan ook ermeld de betalingsvoorwaarden gedeponeerd bij de griffie van de rechtbank te Utrecht onder nummer 12/2013. Indien nog vragen zijn, kunt u altijd contact met ons opnemen via telefoonnummer **0174-622048** of een e-mail sturen naar

(e) $p = 0.887$



Mulder
tandartsen

Declaratie 45316 / 110302

A. AAAAAAAAAAACCHAFBH
Ivige Leane 20
8631WB LOENGA

Code	Element	Aantal	Omschrijving	Bedrag
Betreft behandeling van A. AAAAAAAAAAACCHAFBH, 09-07-1950, 87154997				
Behandelaar - R.S. Tekle, mondhygiënist (13 - AGB: 12053057 - Pr. AGB: 12000000)				
Behandeldatum: 24-2-2017				
M03	-	5	x 5 min. gebitsreiniging hy	61,95
			Totaal:	€ 61,95

G.R. van Staveren-Mulder, tandarts Breedstraat 6, 1381 BX Weesp tel 0294 25 24 14 IBAN NL45INGB0698157001 IBAN NL75INGB0000835004
Gaarne betaling binnen 14 dagen d.m.v. aangehechte acceptgiro. KvK 50614150
15397

deze strook niet meezenden

X 9000045316110302+ 000061954< INGBNL2A+ NL45INGB0698157001+ 17>

voor gebruiksaanwijzing z.o.z.

(f) $p = 0.887$

Figure 8.13: Examples of images incorrectly classified as positive.

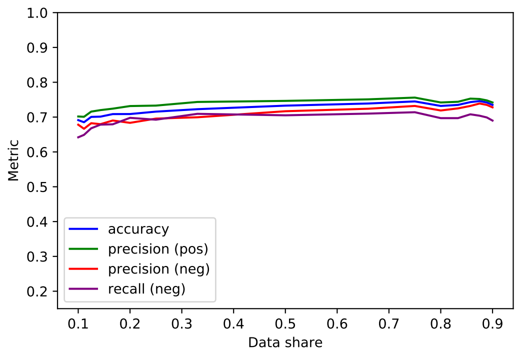
Thus, after looking into these images, we can conclude that most of the negative examples that are classified as positive have some layout or content problem due to which they are classified as negative. These problems are not captured by our classification algorithm and, hence, the images are not classified correctly. Besides this, there are some incorrectly classified cut invoices, hence, some more investigation is required for the correct classification of the cut invoices. Furthermore, we have seen many positive examples with a high predicted probability of belonging to the negative class with no obvious explanation. Thus, it might be possible to find some other features which will still classify well the negative examples but will be able to distinguish the presented positive examples.

8.2.7 Dataset enlargement

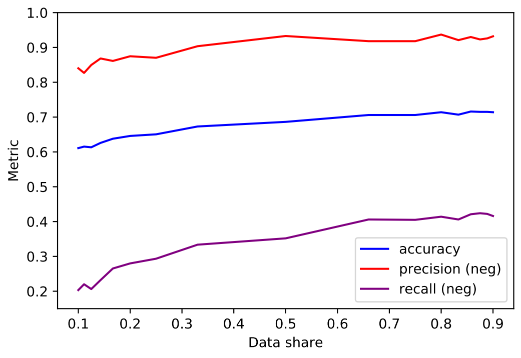
In this section, we will investigate the influence of the extension of the training dataset in different ways on the performance of the classification algorithm. Firstly, the impact of the size of the training dataset on the performance of the algorithm will be discussed. Afterwards, the impact of the inclusion of newly generated blurred images will be discussed.

Training dataset size importance analysis

Firstly, let's have a look at how metrics change when we increase the size of the training dataset. For a random forest classification model with 2 classes and parameters $(400, 15, 3, 2, \text{auto})$ we plot the graphs of the dependence of the evaluation metrics on the dataset share used for training. We consider the models with the thresholds 0.5 (Fig. 8.14a) and 0.3 (Fig. 8.14b). For both models, it does not seem profitable to include more data as the learning process does not give more information starting from around 75%-80% of the dataset. However, the reason for this behaviour might be also the quality of the dataset as we do not have enough negative and partial examples for some image degradations.



(a) The model with the threshold 0.5.



(b) The model with the threshold 0.3.

Figure 8.14: Evaluation metrics depending on the share of the training dataset from the complete dataset for 2 classes random forest classifier.

Training dataset increase with blurred images

In this subsection, we consider another way of increasing the size of the training dataset. We can include in the dataset the blurred versions of fully recognized images that are already present in the dataset.

To generate these images, we blur fully-recognized images with Gaussian blur filters of sizes $5, 7, 9$, and 11 and the variance 100. This will increase the number of negative and partial images. For the larger filter sizes, most of the invoices cannot be recognized by a human neither.

We consider a random forest classification model with 2 classes with the parameters $(400, 15, 3, 2, \text{auto})$. We can include blurred images in the dataset in 2 ways. Firstly, blurred images can be just

Model	Threshold	Accuracy		Precision pos.		Precision neg.	
		mean	std	mean	std	mean	std
Initial	0.5	0.744	0.005	**0.75**	0.004	0.736	0.006
Initial	0.3	0.701	0.005	0.648	0.004	**0.922**	0.009
Mixed blur	0.5	0.683	0.002	0.585	0.003	0.732	0.002
Mixed blur	0.3	0.627	0.004	0.501	0.003	0.842	0.003
Blur linked to original	0.5	**0.752**	0.004	0.723	0.004	0.783	0.004
Blur linked to original	0.3	0.703	0.003	0.627	0.003	0.905	0.005

Table 8.5: Evaluation of the models with included blurred examples.

added into the dataset and the model can be trained with random splits of all data into training and validation sets ("Mixed blur"). However, this might result in having the original image and its blurred versions split into training and validation dataset. Thus, we consider the second method of splitting the data into training and validation datasets. There, we split only original images and, afterwards, include all blurred versions of the corresponding image into the part where the original is located ("Blur linked to original"). In Table 8.5 the evaluation of both models is presented. We can see that in case of the threshold of 0.5 the accuracy and the negative precision improve for the model with blurred versions linked to the original. However, for another model inclusion of blurred versions does not seem profitable.

Thus, the method of enlarging the dataset with blurred images does not seem profitable for the model with the threshold of 0.3. However, for another threshold of 0.5, the dataset extension method with linking blurred images to the originals improves the performance of the model. We have not considered other thresholds and, hence, cannot conclude that this method is not useful in this project. We can only conclude that it does not help the classification with the threshold of 0.3.

Chapter 9

Enhancement techniques

In this chapter, we describe implemented enhancement techniques, their influence on the recognition of images by the ORTEC algorithm, and methods of choosing a relevant enhancement technique to increase the recognition percentage.

9.1 Implemented enhancement techniques

We implement some fast and simple image transformations and spatial and spectral enhancement techniques discussed in Section 4.4. We choose the techniques which are aimed at the sharpening of an image and at skew correction.

We will apply all of the techniques to gray-scaled images in order to shorten and simplify the process. However, they can be also applied to each of the RGB color channels separately in order to preserve colors.

Firstly, we use the binarization transform. We use the respective method from the OpenCV library in Python with adaptive Gaussian thresholding. In this thresholding method, the threshold value for a pixel is defined by the sum of pixel values in a window centered in this pixel and convolved with a Gaussian filter. We use a window of size 31×31 in the experiments.

Secondly, we apply two point-processing transformations: gamma-transformation with $\gamma \in \{0.5, 2\}$ and logarithmic transformation. Moreover, we use two histogram methods: histogram equalization and histogram matching. For the histogram matching, we choose a scan with a sufficient amount of text as an image to which we match others. Local histogram equalization was also implemented but it did not produce images of good quality. These methods can be found in [68].

As an edge enhancement technique, we use the Lapalce method proposed in [90]. We subtract a Laplacian of an image multiplied by a constant $\epsilon \in \{1/21, 1/16, 1/6, 1\}$ from the image itself.

Besides this, the constrained unsharp masking method from [7] is implemented with the Gaussian bilateral filtering (described in [101]) as a smoothing method with $\sigma_r = 20$ and $\sigma_d = 2$. Bilateral filtering is present in the OpenCV Python library. The values of the variance of the unsharp mask used are $\sigma^2 \in \{20, 50\}$. The enhanced image is obtained as the sum of the smoothed image and the masked image multiplied by α with $\alpha \in \{0.2, 0.4\}$.

Furthermore, we implement several spectral methods proposed in [5] and [4] both with DFT and DCT as spectral transforms. As a filtering technique we implement unsharp masking (UM) as in eq. 4.8, modified unsharp masking as in eq. 4.10, alpha rooting (eq. 4.9), and log alpha rooting (eq. 4.11). Modified unsharp masking resulted in images of poor quality with block artifacts (some sort of grid appears on images), thus, this technique was not considered further. In other methods, after some exploration, the following parameters were chosen: $\alpha \in \{0.9, 0.95, 0.975, 0.99\}$ in alpha-rooting, combinations of $(\alpha, \beta, \gamma) \in \{(0.9, 0.7, 1.3), (0.8, 0.7, 0.9), (0.8, 0.9, 1.5)\}$ in log-alpha rooting and $C = 7$ and filters with the variance $\sigma^2 \in \{20, 50\}$ of size 41 in UM.

For skew correction, we try rotating an image by the skew angle obtained from PP, CV, and Sauvola methods. Besides this, we try to rotate each region of the image separately according to its skew angle obtained from PP and CV methods. This method often results in images of poor quality as the text in them becomes discontinuous.

Finally, we apply combinations of enhancement methods that performed well on their own.

9.2 Impact of enhancement techniques

The discussed above enhancement techniques might improve images as well as deteriorate them. That is to say, after the application of any technique, not recognized image can become partially recognized or fully recognized by the ORTEC algorithm and, at the same time, fully recognized images can become partially recognized or not recognized.

We introduce some notation to describe the impact of enhancement techniques. Firstly, let's name the negative and partial images that became positive and negative images that became partial "improved images". Secondly, let's name positive and partial images that became negative and positive images that became partial "worsened images".

9.2.1 Impact of single enhancement techniques

We start by investigating the impact of single enhancement techniques on the images.

The numbers of improved and worsened images for different enhancement techniques are provided in Table 9.1. The total numbers of the positive, partial, and negative examples after application of the respective enhancement techniques can be found in Table 9.2. While evaluating the impact of the enhancement method we should compare it to the results on gray-scale images as all of the enhancement techniques are applied to gray-scaled versions of images. From here we can see, that among the most successful methods there are binarization, constrained UM, DFT and DCT UM, Laplace method, and skew correction methods. We can also see some dependencies between the parameters of enhancement techniques and their impact. For instance, small values of the parameter ϵ for the Laplace method give better results. In the constrained UM the variance σ^2 does not influence the performance while higher α results in both more improvements and more worsenings. For the spectral UM methods the variance σ^2 also does not matter much while the method with DFT used as a spectral transform performs slightly better than the method with DCT. On the other hand, DCT with log-alpha rooting and histogram matching have a much larger negative than positive impact on images. Thus, these enhancement techniques do not add value to the project.

Enhancement method	Worsenings			Improvements		
	From positive		From part.	From negative		From part.
	To part.	To neg.	To neg.	To part.	To pos.	To pos.
Gray-scaling	9	19	11	10	6	20
Binarization	34	26	35	17	**15**	28
Constrained UM $\alpha = 0.2, \sigma^2 = 20$	15	23	28	18	10	29
Constrained UM $\alpha = 0.4, \sigma^2 = 20$	27	22	26	19	12	34
Constrained UM $\alpha = 0.4, \sigma^2 = 50$	27	22	26	19	12	34
DCT alpha rooting $\alpha = 0.9$	32	20	25	**25**	14	16
DCT alpha rooting $\alpha = 0.95$	19	**11**	21	19	10	24
DCT alpha rooting $\alpha = 0.975$	9	12	14	22	11	26
DCT alpha rooting $\alpha = 0.99$	11	12	12	18	8	19
DCT log alpha rooting $(\alpha, \beta, \gamma) = (0.8, 0.7, 0.9)$	32	35	35	16	11	11
DCT log alpha rooting $(\alpha, \beta, \gamma) = (0.8, 0.9, 1.5)$	32	219	104	4	0	0
DCT UM $\sigma^2 = 20$	9	12	11	13	6	19
DCT UM $\sigma^2 = 50$	**7**	14	11	13	6	19
DFT UM $\sigma^2 = 50$	8	14	**9**	15	7	20
Histogram equalization	61	133	60	19	7	7
Laplace $\epsilon = 1/21$	11	15	21	16	3	20
Laplace $\epsilon = 1/16$	9	**11**	16	15	7	23
Laplace $\epsilon = 1/6$	16	16	25	15	6	22
Laplace $\epsilon = 1$	66	118	73	14	3	9
Histograms matching	37	123	82	10	3	8
Log transformation	17	18	24	16	7	23
Gamma transformation $\gamma = 0.5$	32	23	30	22	8	26
Gamma transformation $\gamma = 2$	29	20	26	17	8	18
PP rotation	24	21	18	22	10	25
PP regional rotation + binarization	66	65	50	23	4	22
Sauvola rotation + binarization	36	33	34	20	11	**36**

Table 9.1: Numbers of improved and worsened images for different enhancement techniques.

We can look into some examples of improved and worsened images for different enhancement techniques. We present only the images which improved or worsened in respect to both original and gray-scaled image in order to exclude the influence of conversion to gray-scale. Let's first look into the examples for the methods with not many worsenings and relatively many improvements. In Figure 9.1, the examples after binarization are provided and in Figure 9.2 — the examples for the constrained UM. Sometimes, it is not seen directly from the image itself why it worsens, as in these cases. However, OCR does not recognize any text on the image in Fig. 9.1d, and on the image in Fig. 9.2d, the treatment code and the cost are recognized incorrectly. The examples for the DCT alpha-rooting and log-alpha rooting are given in Figure 9.3. We can notice that the images become darker after the application of these enhancement techniques. Besides this, after the log-alpha rooting some line artifacts

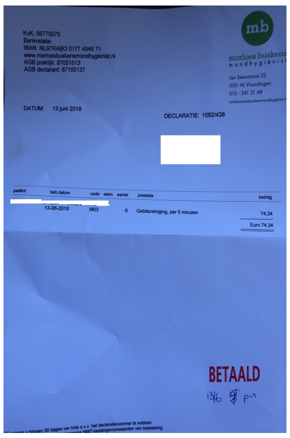
Enhancement method	Totals		
	Positive	Partial	Negative
Original images	314	112	159
Gray-scaling	312	100	173
Binarization	297	100	188
Constrained UM $\alpha = 0.2, \sigma^2 = 20$	315	88	182
Constrained UM $\alpha = 0.4, \sigma^2 = 20$	311	98	176
Constrained UM $\alpha = 0.4, \sigma^2 = 50$	311	98	176
DCT alpha rooting $\alpha = 0.9$	292	128	165
DCT alpha rooting $\alpha = 0.95$	318	105	162
DCT alpha rooting $\alpha = 0.975$	**330**	103	**152**
DCT alpha rooting $\alpha = 0.99$	318	110	157
DCT log alpha rooting $(\alpha, \beta, \gamma) = (0.8, 0.7, 0.9)$	269	114	202
DCT log alpha rooting $(\alpha, \beta, \gamma) = (0.8, 0.9, 1.5)$	63	44	478
DCT UM $\sigma^2 = 20$	318	104	163
DCT UM $\sigma^2 = 50$	318	102	165
DFT UM $\sigma^2 = 50$	319	106	160
Histogtam equalization	134	125	326
Laplace $\epsilon = 1/21$	311	98	176
Laplace $\epsilon = 1/16$	324	97	164
Laplace $\epsilon = 1/6$	310	96	179
Laplace $\epsilon = 1$	142	110	333
Histograms matching	165	69	351
Log transformation	309	98	178
Gamma transformation $\gamma = 0.5$	293	110	182
Gamma transformation $\gamma = 2$	291	114	180
PP rotation	304	115	166
PP regional rotation + binarization	209	129	247
Sauvola rotation + binarization	292	98	195

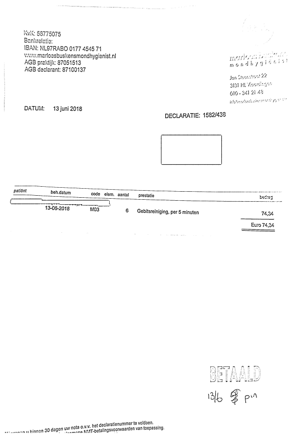
Table 9.2: Total number of positive, partial and negative images in the dataset after application of different enhancement techniques.

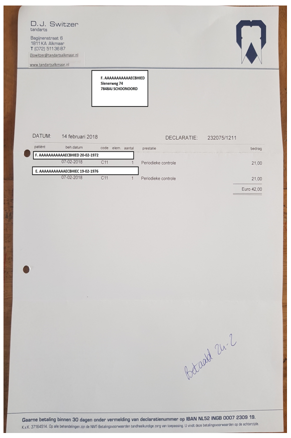
can be observed near the text which results in a worsened image in some cases.

Let's also look into the examples for the methods where we get many worsened images. The examples for histogram methods are given in Figure 9.4. We can see that matching and equalization change the images a lot and, hence, might easily result in a worsening as well as in improvement. The worsenings often happen in the presence of background as in Figures 9.4e-9.4f. For example, for the histogram matching method, the background gets the most amount of dark pixels and, hence, the text on the paper is not that distinguishable.

Thus, we have discussed the impact of different enhancement techniques on image recognition by the OCR algorithm and the ORTEC algorithm. We have concluded that there is no enhancement technique which only improves the images. However, there are some techniques


(a) Original of an improved image.


(b) The improved image.


(c) Original of a worsened image.

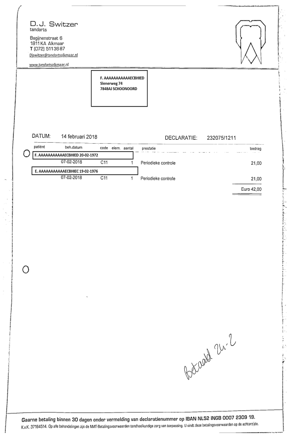
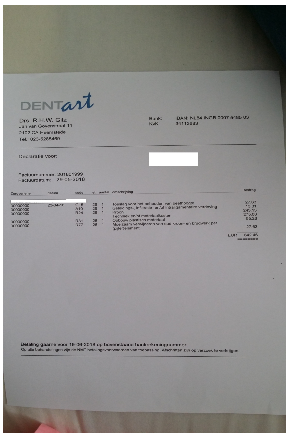
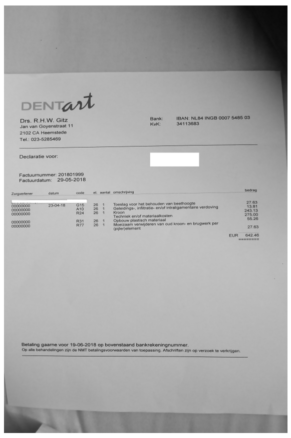
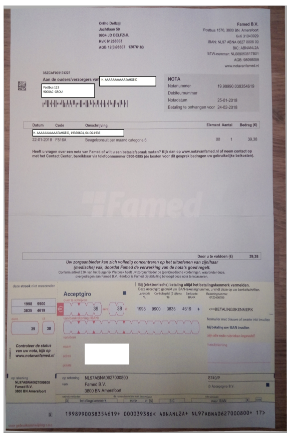

(d) The worsened image.

Figure 9.1: Examples of binarized images.


(a) Original of an improved image.


(b) The improved image.


(c) Original of a worsened image.

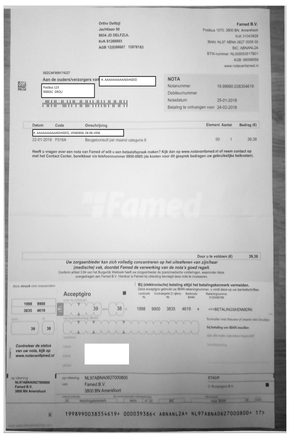

(d) The worsened image.

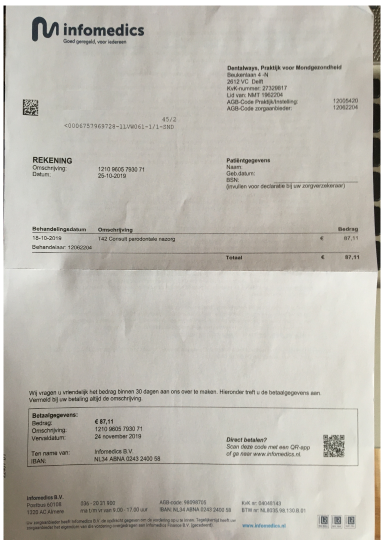
Figure 9.2: Examples of images after constrained UM.

that result in more improvements than worsenings. Among them, there are the Laplace sharpening method, the DCT alpha-rooting method, and the DFT UM method. Some other methods, such as the constrained UM method, result in equally many improvements and worsenings.

9.2.2 Impact of combinations of enhancement techniques

Now, let's look into the impact of some combinations of the enhancement techniques on the invoices recognition process.

We will consider the combinations of the single methods which performed well on their own. Thus, we look at the combinations of the Laplace method with $\epsilon = 1/16$ with the constrained UM with $\alpha = 0.2$ and $\sigma^2 = 20$, DFT UM with $\sigma^2 = 50$, and DCT alpha-rooting with $\alpha = 0.975$. Besides this, we will combine PP rotation with all 4 mentioned sharpening methods.



infomedics
Goed geregeld, voor iedereen

Dentalways, Praktijk voor Mondgezondheid
Beukenlaan 4 -N
2612 VC Delft
KvK-nummer: 27329817
Lid van: NMT 1962204
AGB-Code Praktijk/Instelling: 12005420
AGB-Code zorgaanbieder: 12062204

45/2
<0006757969728-1LVW061-1/1-SND

REKENING
Omschrijving: 1210 9605 7930 71
Datum: 25-10-2019

Patiëntgegevens
Naam:
Geb.datum:
BSN:
(invullen voor declaratie bij uw zorgverzekeraar)

Behandelingsdatum	Omschrijving		Bedrag
18-10-2019	T42 Consult parodontale nazorg	€	87,11
Behandelaar: 12062204			
	Totaal	€	87,11

Wij vragen u vriendelijk het bedrag binnen 30 dagen aan ons over te maken. Hieronder treft u de betaalgegevens aan. Vermeld bij uw betaling altijd de omschrijving.

Betaalgegevens:
Bedrag: € 87,11
Omschrijving: 1210 9605 7930 71
Vervaldatum: 24 november 2019

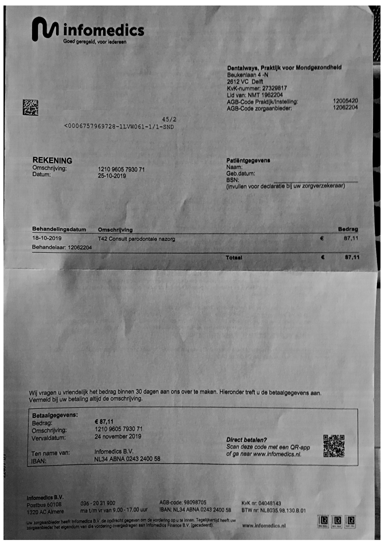
Ten name van: Infomedics B.V.
IBAN: NL34 ABNA 0243 2400 58

Direct betalen?
Scan deze code met een QR-app of ga naar www.infomedics.nl.

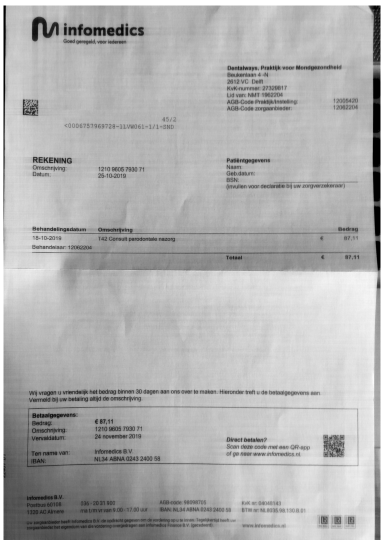
Infomedics B.V.
Postbus 60108
1320 AC Almere
036 - 20 31 900
ma t/m vr van 9.00 - 17.00 uur
AGB-code: 98098705
IBAN: NL34 ABNA 0243 2400 58
KvK nr: 04048143
BTW nr: NL8035.98.130.B.01

Uw zorgaanbieder heeft Infomedics B.V. de opdracht gegeven om de vordering op u te innen. Tegelijkertijd heeft uw zorgaanbieder het eigendom van die vordering overgedragen aan Infomedics Finance B.V. (gecedeerd).
www.infomedics.nl

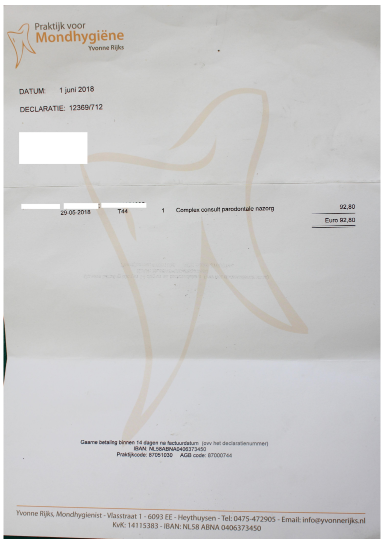
(a) Original of an improved image.



(b) The improved image after DCT alpha rooting.



(c) The improved image after DCT log-alpha rooting.



Praktijk voor
Mondhygiëne
Yvonne Rijks

DATUM: 1 juni 2018

DECLARATIE: 12369/712

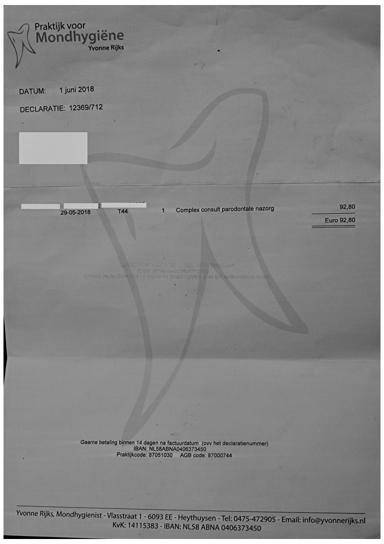
[redacted]

[redacted] [redacted] [redacted]
29-05-2018 T44 1 Complex consult parodontale nazorg 92,80
Euro 92,80

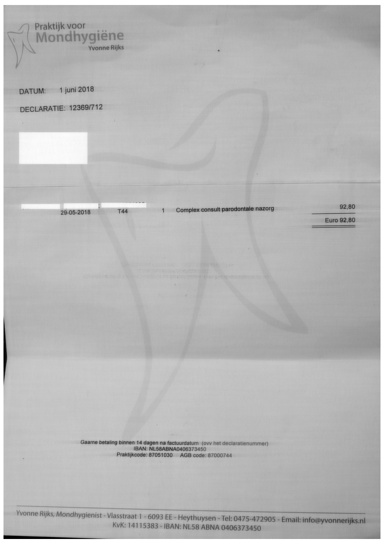
Gaarne betaling binnen 14 dagen na factuurdatum (ovv het declaratienummer)
IBAN: NL58ABNA0406373450
Praktijkcode: 87051030 AGB code: 87000744

Yvonne Rijks, Mondhygienist - Vlasstraat 1 - 6093 EE - Heythuysen - Tel: 0475-472905 - Email: info@yvonnerijks.nl
KvK: 14115383 - IBAN: NL58 ABNA 0406373450

(d) Original of a worsened image.



(e) The worsened image after DCT alpha rooting.

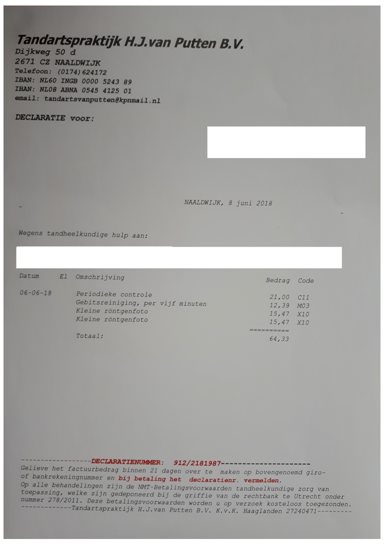


(f) The worsened image after DCT log-alpha rooting.

Figure 9.3: Examples of improved and worsened images after DCT alpha-rooting and log-alpha rooting.

In addition, we binarize the resulted images. The complete results with a comparison with the outcome of the same enhancement techniques used on their own can be found in Table B.1 in Appendix B.

We can see that combining the Laplace and the constrained UM methods improves the results which were obtained by only one of these 2 techniques. We can also notice the slight differences between different order of methods application. Other pairs of sharpening methods do not outperform sharpening methods used on their own. Applying binarization after combinations of sharpening techniques does not improve the results either. On the other hand, the rotation method performs better when combined with a sharpening method, as, in this case,



Tandartspraktijk H.J.van Putten B.V.
Dijkweg 50 d
2671 CZ NAALDWIJK
Telefoon: (0174)624172
IBAN: NL60 INGB 0000 5243 89
IBAN: NL08 ABNA 0545 4125 01
email: tandartsvanputten@kpnmail.nl

DECLARATIE voor:

[redacted]

NAALDWIJK, 8 juni 2018

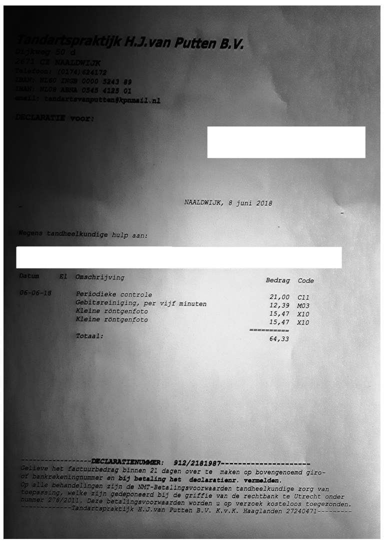
Wegens tandheelkundige hulp aan:

[redacted]

Datum	El	Omschrijving	Bedrag	Code
06-06-18		Periodieke controle	21,00	C11
		Gebitsreiniging, per vijf minuten	12,39	M03
		Kleine röntgenfoto	15,47	X10
		Kleine röntgenfoto	15,47	X10
			==========	
		Totaal:	64,33	

-----------------DECLARATIENUMMER: 912/2181987---------------------
Gelieve het factuurbedrag binnen 21 dagen over te maken op bovengenoemd giro-
of bankrekeningnummer en **bij betaling het declaratienr. vermelden.**
Op alle behandelingen zijn de NMT-Betalingsvoorwaarden tandheelkundige zorg van
toepassing, welke zijn gedeponeerd bij de griffie van de rechtbank te Utrecht onder
nummer 278/2011. Deze betalingsvoorwaarden worden u op verzoek kosteloos toegezonden.
------------Tandartspraktijk H.J.van Putten B.V. K.v.K. Haaglanden 27240471---------

(a) Original of an improved image.



Tandartspraktijk H.J.van Putten B.V.
Dijkweg 50 d
2671 CZ NAALDWIJK
Telefoon: (0174)624172
IBAN: NL60 INGB 0000 5243 89
IBAN: NL08 ABNA 0545 4125 01
email: tandartsvanputten@kpnmail.nl

DECLARATIE voor:

[redacted]

NAALDWIJK, 8 juni 2018

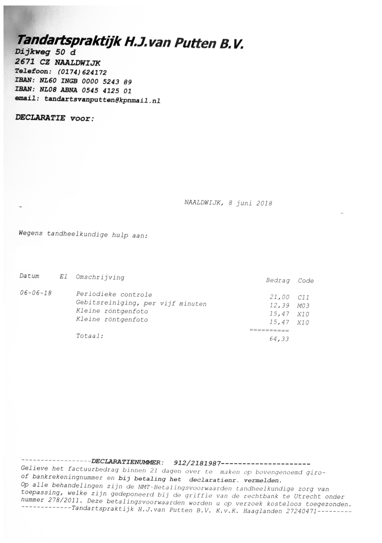
Wegens tandheelkundige hulp aan:

[redacted]

Datum El Omschrijving Bedrag Code
06-06-18 Periodieke controle 21,00 C11
Gebitsreiniging, per vijf minuten 12,39 M03
Kleine röntgenfoto 15,47 X10
Kleine röntgenfoto 15,47 X10
==========
Totaal: 64,33

-----------------DECLARATIENUMMER: 912/2181987---------------------
Gelieve het factuurbedrag binnen 21 dagen over te maken op bovengenoemd giro-
of bankrekeningnummer en bij betaling het declaratienr. vermelden.
Op alle behandelingen zijn de NMT-Betalingsvoorwaarden tandheelkundige zorg van
toepassing, welke zijn gedeponeerd bij de griffie van de rechtbank te Utrecht onder
nummer 278/2011. Deze betalingsvoorwaarden worden u op verzoek kosteloos toegezonden.
------------Tandartspraktijk H.J.van Putten B.V. K.v.K. Haaglanden 27240471---------

(b) The improved image after histogram equalization.



Tandartspraktijk H.J.van Putten B.V.
Dijkweg 50 d
2671 CZ NAALDWIJK
Telefoon: (0174)624172
IBAN: NL60 INGB 0000 5243 89
IBAN: NL08 ABNA 0545 4125 01
email: tandartsvanputten@kpnmail.nl

DECLARATIE voor:

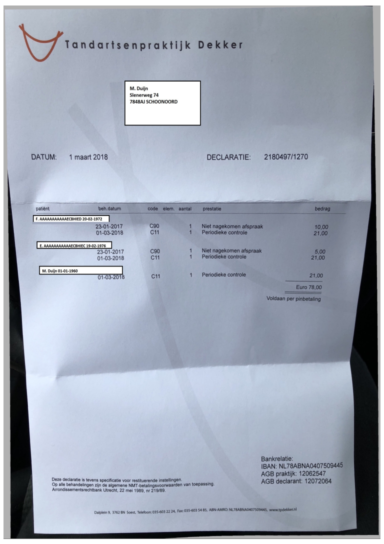
NAALDWIJK, 8 juni 2018

Wegens tandheelkundige hulp aan:

Datum El Omschrijving Bedrag Code
06-06-18 Periodieke controle 21,00 C11
Gebitsreiniging, per vijf minuten 12,39 M03
Kleine röntgenfoto 15,47 X10
Kleine röntgenfoto 15,47 X10
==========
Totaal: 64,33

-----------------DECLARATIENUMMER: 912/2181987---------------------
Gelieve het factuurbedrag binnen 21 dagen over te maken op bovengenoemd giro-
of bankrekeningnummer en bij betaling het declaratienr. vermelden.
Op alle behandelingen zijn de NMT-Betalingsvoorwaarden tandheelkundige zorg van
toepassing, welke zijn gedeponeerd bij de griffie van de rechtbank te Utrecht onder
nummer 278/2011. Deze betalingsvoorwaarden worden u op verzoek kosteloos toegezonden.
------------Tandartspraktijk H.J.van Putten B.V. K.v.K. Haaglanden 27240471---------

(c) The improved image after histogram matching.



Tandartsenpraktijk Dekker

M. Duijn
Slenerweg 74
7848AJ SCHOONOORD

DATUM: 1 maart 2018 DECLARATIE: 2180497/1270

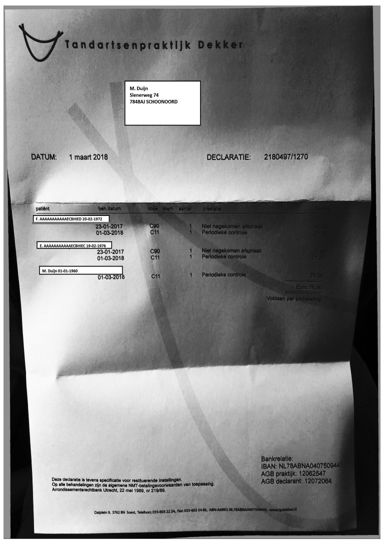
patiënt	beh.datum	code	elem.	aantal	prestatie	bedrag
F. AAAAAAAAAAAECBHIED 20-02-1972						
	23-01-2017	C90		1	Niet nagekomen afspraak	10,00
	01-03-2018	C11		1	Periodieke controle	21,00
E. AAAAAAAAAAAECBHIEC 19-02-1976						
	23-01-2017	C90		1	Niet nagekomen afspraak	5,00
	01-03-2018	C11		1	Periodieke controle	21,00
M. Duijn 01-01-1960						
	01-03-2018	C11		1	Periodieke controle	21,00
						Euro 78,00
						Voldaan per pinbetaling

Bankrelatie:
IBAN: NL78ABNA0407509445
AGB praktijk: 12062547
AGB declarant: 12072064

Deze declaratie is tevens specificatie voor restituerende instellingen.
Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing.
Arrondissementsrechtbank Utrecht, 22 mei 1989, nr 219/89.

Dalplein 9, 3762 BN Soest, Telefoon: 035-603 22 24, Fax: 035-603 54 85, ABN-AMRO: NL78ABNA0407509445, www.tpdekker.nl

(d) Original of a worsened image.



Tandartsenpraktijk Dekker

M. Duijn
Slenerweg 74
7848AJ SCHOONOORD

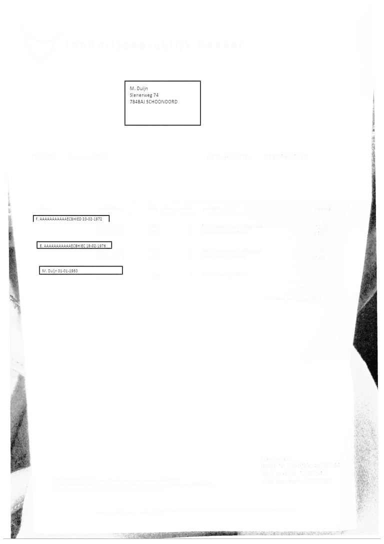
DATUM: 1 maart 2018 DECLARATIE: 2180497/1270

patiënt beh.datum code elem. aantal prestatie bedrag
F. AAAAAAAAAAAECBHIED 20-02-1972
23-01-2017 C90 1 Niet nagekomen afspraak
01-03-2018 C11 1 Periodieke controle
E. AAAAAAAAAAAECBHIEC 19-02-1976
23-01-2017 C90 1 Niet nagekomen afspraak
01-03-2018 C11 1 Periodieke controle
M. Duijn 01-01-1960
01-03-2018 C11 1 Periodieke controle
Voldaan per pinbetaling

Bankrelatie:
IBAN: NL78ABNA0407509445
AGB praktijk: 12062547
AGB declarant: 12072064

Deze declaratie is tevens specificatie voor restituerende instellingen.
Op alle behandelingen zijn de algemene NMT-betalingsvoorwaarden van toepassing.
Arrondissementsrechtbank Utrecht, 22 mei 1989, nr 219/89.

(e) The worsened image after histogram equalization.



M. Duijn
Slenerweg 74
7848AJ SCHOONOORD

F. AAAAAAAAAAAECBHIED 20-02-1972
E. AAAAAAAAAAAECBHIEC 19-02-1976
M. Duijn 01-01-1960

(f) The worsened image after histogram matching.

Figure 9.4: Examples of improved and worsened images after histograms equalization and matching.

the technique corrects for both skew and blur. However, overall, we do not see significant improvements compared to the results obtained before: the number of positive examples after application of the DCT alpha-rooting with $\alpha = 0.975$ is similar to the maximum number of positive examples obtained after application of combinations of enhancement techniques as can be seen in Table B.2. The maximum number of positive examples is obtained after application of the combination of the Laplace transform and constrained UM.

Thus, we can conclude that the combinations of the sharpening enhancement techniques do not outperform the single sharpening enhancement techniques. On the other hand, combining the techniques which correct for different degradations can be useful as in the case of combination of sharpness improvement and deskewing techniques.

9.3 Computational complexity of enhancement techniques

In Table 9.3 computational complexity of different image enhancement algorithms for several image sizes is given. Calculation time for rotation algorithms can be reduced as we will already know the angles for rotation after feature extraction. Besides this, DFT and DCT of the image can be calculated only once and might be calculated already in the feature extraction process. Hence, the respective enhancement times could be also reduced.

We can see that the enhancement techniques which involve spectral transforms take around 1 second or even more to perform for the average-sized images. Other enhancement techniques are less computationally complex.

Thus, some of the enhancement techniques are too slow to be applied to the original image. Hence, they can be applied only to a downscaled image. However, the outcome of this enhancement might differ from the outcome on the original image and this method requires further studies.

Image size / Feature	1200 × 900		2500 × 1900		4000 × 3000	
	mean	std	mean	std	mean	std
Constrained unsharp masking	0.400	0.018	1.800	0.157	4.300	0.141
Binarization	0.011	0.002	0.046	0.003	0.115	0.007
Logarithm transform	0.043	0.003	0.152	0.011	0.384	0.016
Power transform	0.038	0.016	0.150	0.077	0.380	0.193
Histogram equalization	0.049	0.004	0.166	0.013	0.415	0.024
Histogram matching	0.089	0.006	0.253	0.016	0.593	0.030
Laplace transform	0.029	0.005	0.112	0.006	0.285	0.012
DFT with alpha-rooting	0.189	0.011	0.910	0.040	2.293	0.077
DCT with alpha-rooting	0.176	0.059	0.866	0.250	2.235	0.633
DFT with log alpha-rooting	0.323	0.021	1.521	0.117	3.838	0.013
DCT with log alpha-rooting	0.250	0.016	1.217	0.042	3.124	0.092
DFT with unsharp masking	0.256	0.023	1.287	0.045	3.097	0.116
DCT with unsharp masking	0.209	0.071	1.047	0.034	2.738	0.079
PP full rotation	0.025	0.004	0.076	0.008	0.167	0.015
PP regional rotation	0.012	0.004	0.052	0.015	0.148	0.046
CV full rotation	0.025	0.005	0.105	0.012	0.272	0.020
CV regional rotation	0.012	0.004	0.049	0.014	0.139	0.043
Sauvola full rotation	0.013	0.002	0.052	0.006	0.128	0.014

Table 9.3: Computational complexity for image enhancement algorithms (in seconds) for different image sizes.

9.4 Classification algorithms for choosing relevant enhancement technique

We have seen in Section 9.2 that image quality can deteriorate as well as improve after the application of different techniques. Our goal is to understand which enhancement technique to apply to which image in order to maximize the percentage of images recognized by the ORTEC algorithm.

Thus, to achieve this goal, a classification algorithm can be constructed based on the outcomes of the ORTEC algorithm for images after the application of enhancement methods. As the independent variables, we can use the same features as in Chapter 8. We will use 189 uncorrelated features given in Section 8.2.3.

However, we have to introduce a new labelling for the new problem. For each of the enhancement techniques we can introduce several labellings:

- 1 for improved images and 0 for all other images;
- 0 for worsened images and 1 for all other images;
- 2 for improved, 0 for worsened, and 1 for all other images.

The next step is to choose the evaluation metrics. As there is quite a small number of the improved images, as we can see in Tables 9.1 and B.1, the accuracy metric is not relevant in this classification. A greater role will be played by the recall (eq. 9.1). Depending on the labelling, it will show one or both things out of the following:

- For how many images out of those that can be improved by a certain enhancement technique we actually will choose to apply this technique;
- For how many images out of those that are worsened by a certain enhancement technique we will choose to apply this technique and, actually, worsen them.

$$recall_{\text{class } i} = \frac{\#\text{ correctly classified images of class } i}{\#\text{ images in class } i}. \tag{9.1}$$

Precision will be still important as well. It will allow understanding which percentage of the images which were chosen for enhancement will be actually improved and which will be worsened, depending on the labelling and a class for which precision is calculated.

As a classification algorithm, we use a random forest model as well as in Chapter 8. For the classification into 2 classes, we use a random forest classifier and for classification into 3 classes, we use a random forest regressor. We consider thresholds of 0.3 and 0.5 for 2 classes classification and threshold pairs of 0.5|1.5 and 0.3|1.5 for 3 classes classification.

9.5 Evaluation of classification algorithms

In this section, we will evaluate random forest algorithms for a choice of enhancement techniques with different parameters and for different data labellings from 9.4.

Enhancement method	Maximizing, Labelling	Recall pos.		Recall neg.		Precision pos.		Precision neg.	
		mean	std	mean	std	mean	std	mean	std
Constrained UM, thr 0.3	Recall pos, 1	0.145	0.017	0.928	0.004	0.176	0.018	0.911	0.002
Sauvola rotation + binarization	Precision pos, 1	0.031	0.004	0.996	0.002	0.497	0.111	0.888	0.001
DCT log-alpha rooting	Recall neg, 2	0.67	0.008	0.694	0.006	0.673	0.004	0.691	0.005
Equalization	2	0.8	0.004	0.549	0.005	0.687	0.003	0.689	0.006
DFT UM	Recall pos, 2	0.991	0.001	0	0	0.934	0.001	0	0
Binarization, thr 0.3—1.5	Recall pos, 3	0.024	0.008	0.042	0.009	0.85	0.229	0.22	0.043
DCT log-alpha rooting, thr 0.3—1.5	Recall neg, 3	0.008	0.017	0.849	0.006	0.1	0.2	0.559	0.004

Table 9.4: Evaluation metrics for classification of impact of enhancement techniques on recognition.

The evaluation of these classification algorithms shows that they do not perform well enough for any labelling. Some of the best results for each type of labelling according to the respective important metrics (positive or negative recall and positive or negative precision) are given in Table 9.4.

For the first labelling we are interested in the positive recall and the positive precision and we can see that both metrics in the best models have low values. For the second labelling we are interested in the negative recall. We get reasonable values for the enhancement techniques with a large number of worsenings, such as DCT log-alpha rooting or histogram equalization. However, we have seen in Section 9.2.1 that these methods result in a few improvements and are not helpful for image enhancement. On the other hand, when it comes to other methods with fewer worsenings we do not get the negative recall higher than 0.11. This means that if we follow that classification, we will apply the enhancement technique to many images which will worsen due to it. For the third labelling we are interested in both recall values. However, the maximum value of the positive recall is very small. The maximum value of the negative recall is obtained for the techniques with a large number of worsened images and the positive recall and precision are very small in these cases. Thus, all 3 labellings do not provide high enough precision and recall in the random forest classification and regression models.

Thus, we have investigated different enhancement techniques and found several of them that might improve image quality. However, we could not find a classification algorithm that will accurately predict if the image will be recognized or not after application of these enhancement techniques.

9.6 Other possible approaches to image enhancement

In Section 9.5 we have not found a classification algorithm that will help to choose the enhancement technique which will improve the image. Therefore, we can think of other approaches to image enhancement where we do not need to perform classification.

One such approach is to apply some image enhancement techniques to the image only if it is predicted as negative by the prediction algorithm from Chapter 8. As we have seen that only a few images improve after application of any enhancement technique (Sec. 9.2), we cannot run the ORTEC algorithm immediately after enhancement as it can still result in a not recognized image. Therefore, we can run the prediction algorithm again on the enhanced image. This will take slightly more than a second according to Table 8.4 in Section 8.2.5. Thus, the whole procedure will take around 3 seconds: firstly, predict the class, then run an enhancement technique, and, finally, predict the class again. Thus, this approach cannot be applied in practice as it takes too long to run.

To sum up, we have investigated several different approaches to image enhancement. However, none of them can be implemented in practice as they do not provide a significant improvement of the recognition process or take a long time to run.

Chapter 10

Future research

In this project, we have developed an image classification algorithm and have investigated various image enhancement methods. However, there is still a large area for the research remaining. In this chapter, we discuss possible directions for future research.

Let's start by discussing possible research directions for the classification problem. Firstly, other features can be added to the classification model. For example, other methods suggested in Sections 4.1-4.3.1 can be used for feature extraction. Secondly, some currently used features can be substituted by correlated ones which are less computationally expensive. This way, for example, the Laplace sharpness and FISH sharpness (Sec. 7.1) can be used instead of the respective simple IL-NIQE Gabor features they are correlated with. Thirdly, some features can be calculated only for the middle of the image to avoid problems with background. For example, contrast features are very sensitive to the background but if they are evaluated only in the middle of the image, they should focus on the contrast of the invoice itself.

In addition, it might be considered to use not only image features but also the text extracted by means of OCR and include it as a feature for the algorithm. However, this is already touching upon problems related to the content of an invoice. Besides this, it depends on the speed of OCR API if it is worth including this information in the algorithm.

The next possible point of the improvement of the classification algorithm can be the choice of another machine learning algorithm instead of a random forest. For instance, SVM, logistic regression, or k-NN can be investigated more thoroughly.

Furthermore, a more optimal combination of the currently extracted features can be discovered for the classification problem. There is a broad range of available options in terms of the extracted features and sizes of images which to use for extraction. This way, both the computational complexity and the quality of the model can be improved.

Finally, the implementations of the feature extraction algorithms can be suboptimal in this project and, hence, can be further improved.

The second problem approached in this project is the image enhancement problem. No universal image enhancement technique was discovered. Thus, other techniques suggested in Section 4.4 can be implemented for image enhancement. Moreover, suitable enhancement techniques for text images can be developed.

New techniques could result in a higher percentage of the images recognized after enhancement. In addition, a better algorithm for classifying the images depending on the enhancement technique that improves the image can be developed.

One more possible approach for research is to apply image enhancement techniques only to the text regions. For it, text localization could be performed at first, then, the text regions can be corrected and deskewed and passed to the OCR API.

Thus, we can see that there are many directions in which the research of the problems set in this project can be continued.

Chapter 11

Conclusions

In the project, we have approached the problems of text images quality assessment and enhancement subject to time constraints. As the examples of text images, invoices from the dental care were provided by ORTEC, the company where this project was performed. As a subjective quality metric of an image, an outcome of the ORTEC algorithm (Chapter 3) was used.

During the course of the project, the existing algorithm and the available data were carefully studied. Furthermore, a literature research in computer vision was performed in order to get familiar with the image analysis field. Finally, random forest classification models were applied and investigated.

An algorithm to assess the quality of text images was suggested based on feature extraction and random forest classification. The size of the available dataset did not allow for use of deep learning frameworks and that is the reason why features were extracted to perform learning. After extensive literature research on image statistics extraction and image quality assessment, the features were chosen and extracted. The further analysis helped to select the features which balanced the selected quality measures and computational complexity. Finally, the whole framework was developed for text image quality assessment, more specifically, invoices images quality assessment.

This framework consists of three steps. Firstly, we downscale the input image to 4 different sizes, 1200×900, 800×600, 600×500, and 400×300. Then, the features are extracted from the downscaled images or the original image: for each feature the size of the image to extract it is chosen depending on the probability threshold that is used in the model. After the feature extraction, the random forest classifier is run that returns the probability of the image belonging to the positive class. If this probability is larger than the threshold, the image is classified as positive and passed to the ORTEC algorithm. Otherwise, it is classified as negative and a user is asked to provide a new, higher quality photo.

In the end, 3 random forest classification models were suggested for the classification of invoices images into 2 classes. The considered probability thresholds are 0.2, 0.3, and 0.4. The larger is the threshold, the lower is the obtained negative precision and the higher is the recall. The models result in the estimated precision values of 0.94, 0.91, and 0.83 and recall values 0.25, 0.43, and 0.58 respectively. Thus, depending on the business needs and acceptable trade-off between these metrics, one of the models can be used.

Each of the models takes less than 1.4 seconds to run and involves the extraction of 34 (in 2 of them) or 32 features (in the remaining one with the threshold 0.4).

The second problem which was considered in the project is text image enhancement. After literature research of existing enhancement techniques, we have implemented some of them. As these methods not only improved some images but also degraded others, we tried to perform classification based on the features extracted for the previous problem using random forests. Using the classifier, we aimed at separating the improved enhanced images from the degraded ones. However, this classification did not give valuable results. We also suggested another possible approach using the already developed classification algorithm for predicting the influence of enhancement methods. This method was too computationally expensive for the use-case and, hence, was not investigated in detail.

To sum up, the project resulted in the fast text image quality classification algorithm which includes a feature extraction step and a classification step. Moreover, image enhancement techniques were studied and their performance on text images was investigated.

Appendix A

Use of downscaled images for features extraction

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain
		mean	std	mean	std	mean	std	(sec)
None (original model)		0.762	0.003	0.841	0.006	0.602	0.005	0.000
S3 sharpness	ds1	0.758	0.004	0.836	0.005	0.597	0.007	0.703
S3 sharpness	ds2	0.757	0.002	0.832	0.005	0.598	0.005	0.768
S3 sharpness	ds3	0.757	0.003	0.834	0.006	0.596	0.004	0.786
Brightness	ds1	0.764	0.004	0.84	0.007	0.608	0.005	0.007
Brightness	ds2	0.761	0.004	0.84	0.007	0.603	0.006	0.017
Brightness	ds3	0.763	0.004	0.841	0.005	0.607	0.007	0.019
Contrast	ds1	0.761	0.006	0.841	0.006	0.601	0.008	0.129
Contrast	ds2	0.762	0.004	0.841	0.007	0.604	0.006	0.141
Contrast	ds3	0.762	0.004	0.84	0.006	0.604	0.006	0.144
Borders morphological	ds1	0.762	0.003	0.835	0.005	0.609	0.008	1.169
Borders morphological	ds2	0.757	0.003	0.832	0.006	0.599	0.005	1.276
Borders morphological	ds3	0.758	0.004	0.833	0.006	0.601	0.006	1.307
PP	ds1	0.76	0.005	0.834	0.007	0.605	0.008	0.247
PP	ds2	0.76	0.005	0.838	0.007	0.601	0.009	0.275
PP	ds3	0.763	0.002	0.841	0.004	0.606	0.004	0.283
CV	ds1	0.761	0.004	0.838	0.005	0.605	0.008	0.209
CV	ds2	0.763	0.002	0.843	0.005	0.603	0.005	0.232
CV	ds3	0.764	0.004	0.838	0.007	0.61	0.007	0.239
S3 sharpness regional	ds1	0.764	0.003	0.841	0.007	0.609	0.005	0.323
S3 sharpness regional	ds2	0.763	0.005	0.842	0.009	0.605	0.006	0.351
S3 sharpness regional	ds3	0.765	0.004	0.841	0.006	0.61	0.005	0.359
BRISQUE	ds1	0.754	0.005	0.825	0.007	0.599	0.009	0.865
BRISQUE	ds2	0.75	0.005	0.818	0.007	0.596	0.008	0.983
BRISQUE	ds3	0.751	0.004	0.823	0.005	0.593	0.008	1.017
S index	ds1	0.766	0.005	0.847	0.009	0.606	0.006	3.788
S index	ds2	0.765	0.004	0.844	0.007	0.608	0.007	4.093

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain
		mean	std	mean	std	mean	std	(sec)
S index	ds3	0.769	0.004	0.848	0.009	0.614	0.006	4.180
Simple Gabor	ds1	0.765	0.005	0.848	0.006	0.603	0.007	8.587
Simple Gabor	ds2	0.764	0.004	0.846	0.008	0.604	0.006	9.327
Simple Gabor	ds3	0.763	0.004	0.847	0.004	0.599	0.008	9.533
Gradient Gabor	ds1	0.766	0.004	0.846	0.007	0.61	0.006	4.463
Gradient Gabor	ds2	0.765	0.004	0.842	0.005	0.611	0.007	4.848
Gradient Gabor	ds3	0.765	0.004	0.84	0.004	0.61	0.006	4.955
Gradient IL-NIQE	ds1	0.765	0.003	0.838	0.005	0.614	0.005	2.798
Gradient IL-NIQE	ds2	0.767	0.004	0.845	0.007	0.612	0.007	3.053
Gradient IL-NIQE	ds3	0.764	0.003	0.843	0.005	0.606	0.007	3.126
All features	ds1	0.729	0.006	0.822	0.008	0.535	0.01	20.114
All features	ds2	0.723	0.004	0.803	0.008	0.536	0.009	21.907
All features	ds3	0.717	0.002	0.774	0.006	0.554	0.004	22.415

Table A.1: Evaluation of the models with the threshold 0.4 and 25 features when extracting specified features from downscaled images.

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain
		mean	std	mean	std	mean	std	(sec)
None (original model)		0.737	0.006	0.919	0.009	0.476	0.01	0.000
S3 sharpness	ds1	0.738	0.003	0.918	0.005	0.481	0.006	0.703
S3 sharpness	ds2	0.734	0.004	0.911	0.01	0.475	0.007	0.768
S3 sharpness	ds3	0.734	0.005	0.914	0.009	0.474	0.008	0.786
Brightness	ds1	0.74	0.004	0.918	0.007	0.484	0.006	0.007
Brightness	ds2	0.741	0.004	0.927	0.008	0.482	0.005	0.017
Brightness	ds3	0.741	0.004	0.926	0.008	0.483	0.007	0.019
Contrast	ds1	0.736	0.005	0.917	0.008	0.476	0.01	0.129
Contrast	ds2	0.735	0.004	0.919	0.004	0.473	0.008	0.141
Contrast	ds3	0.737	0.003	0.923	0.004	0.475	0.007	0.144
Borders morphological	ds1	0.737	0.004	0.919	0.005	0.477	0.009	1.169
Borders morphological	ds2	0.736	0.003	0.918	0.006	0.476	0.006	1.276
Borders morphological	ds3	0.735	0.002	0.916	0.006	0.475	0.004	1.307
PP	ds1	0.738	0.002	0.918	0.005	0.48	0.005	0.247
PP	ds2	0.738	0.005	0.919	0.007	0.479	0.008	0.275
PP	ds3	0.737	0.004	0.918	0.005	0.477	0.008	0.283
CV	ds1	0.736	0.004	0.916	0.006	0.476	0.007	0.209
CV	ds2	0.735	0.004	0.916	0.01	0.475	0.006	0.232
CV	ds3	0.738	0.004	0.92	0.008	0.479	0.007	0.239
S3 sharpness regional	ds1	0.73	0.005	0.899	0.009	0.473	0.009	0.323
S3 sharpness regional	ds2	0.733	0.005	0.902	0.008	0.479	0.009	0.351
S3 sharpness regional	ds3	0.73	0.003	0.902	0.007	0.471	0.009	0.359
Laplace sharpness regional	ds1	0.736	0.003	0.918	0.004	0.475	0.006	0.102
Laplace sharpness regional	ds2	0.737	0.003	0.92	0.006	0.476	0.007	0.11
Laplace sharpness regional	ds3	0.736	0.003	0.923	0.007	0.474	0.007	0.113
BRISQUE	ds1	0.737	0.004	0.914	0.005	0.479	0.008	0.865
BRISQUE	ds2	0.738	0.003	0.92	0.008	0.479	0.007	0.983
BRISQUE	ds3	0.737	0.004	0.922	0.007	0.476	0.007	1.017
Morphology	ds1	0.737	0.005	0.922	0.005	0.476	0.009	0.765
Morphology	ds2	0.736	0.004	0.917	0.008	0.478	0.007	0.785
Morphology	ds3	0.736	0.004	0.917	0.008	0.477	0.008	0.791
S index	ds1	0.735	0.004	0.92	0.004	0.473	0.008	3.788
S index	ds2	0.735	0.004	0.92	0.009	0.473	0.006	4.093
S index	ds3	0.736	0.004	0.923	0.008	0.472	0.008	4.180
Simple Gabor	ds1	0.733	0.004	0.928	0.007	0.462	0.009	8.587
Simple Gabor	ds2	0.729	0.003	0.927	0.006	0.455	0.007	9.327
Simple Gabor	ds3	0.728	0.005	0.931	0.008	0.45	0.009	9.533
Gradient Gabor	ds1	0.731	0.005	0.914	0.007	0.467	0.008	4.463
Gradient Gabor	ds2	0.732	0.006	0.917	0.009	0.466	0.009	4.848
Gradient Gabor	ds3	0.732	0.005	0.915	0.007	0.468	0.011	4.955
IL-NIQE colors	ds1	0.738	0.003	0.926	0.008	0.475	0.005	1.459
IL-NIQE colors	ds2	0.739	0.005	0.919	0.006	0.483	0.01	1.594
IL-NIQE colors	ds3	0.738	0.004	0.92	0.007	0.479	0.008	1.634

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain (sec)
		mean	std	mean	std	mean	std	
Gradient IL-NIQE	ds1	0.736	0.005	0.912	0.007	0.479	0.009	2.798
Gradient IL-NIQE	ds2	0.734	0.003	0.91	0.006	0.477	0.006	3.053
Gradient IL-NIQE	ds3	0.736	0.002	0.914	0.008	0.478	0.005	3.126
All features	ds1	0.696	0.003	0.884	0.007	0.399	0.007	25.683
All features	ds2	0.692	0.005	0.876	0.013	0.396	0.006	27.924
All features	ds3	0.686	0.003	0.86	0.005	0.39	0.007	28.563

Table A.2: Evaluation of the models with the threshold 0.3 and 50 features when extracting specified features from downscaled images.

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain
		mean	std	mean	std	mean	std	(sec)
None (original model)		0.675	0.004	0.968	0.008	0.312	0.009	0.000
S3 sharpness	ds1	0.681	0.003	0.968	0.007	0.326	0.007	0.703
S3 sharpness	ds2	0.676	0.005	0.964	0.006	0.316	0.011	0.768
S3 sharpness	ds3	0.674	0.004	0.964	0.008	0.312	0.006	0.786
Brightness	ds1	0.676	0.005	0.965	0.01	0.315	0.01	0.007
Brightness	ds2	0.676	0.005	0.967	0.008	0.315	0.011	0.017
Brightness	ds3	0.677	0.004	0.969	0.007	0.317	0.009	0.019
Contrast	ds1	0.677	0.004	0.971	0.007	0.317	0.008	0.129
Contrast	ds2	0.674	0.004	0.967	0.006	0.311	0.009	0.141
Contrast	ds3	0.675	0.002	0.969	0.008	0.312	0.005	0.144
Borders morphological	ds1	0.677	0.003	0.964	0.004	0.318	0.006	1.169
Borders morphological	ds2	0.677	0.006	0.963	0.013	0.318	0.011	1.276
Borders morphological	ds3	0.676	0.003	0.96	0.007	0.318	0.006	1.307
PP	ds1	0.679	0.004	0.975	0.005	0.318	0.007	0.247
PP	ds2	0.679	0.004	0.974	0.005	0.32	0.008	0.275
PP	ds3	0.68	0.003	0.969	0.008	0.323	0.006	0.283
CV	ds1	0.677	0.004	0.967	0.006	0.317	0.007	0.209
CV	ds2	0.676	0.004	0.966	0.009	0.316	0.008	0.232
CV	ds3	0.68	0.004	0.975	0.005	0.321	0.007	0.239
S3 sharpness regional	ds1	0.674	0.005	0.959	0.009	0.313	0.009	0.323
S3 sharpness regional	ds2	0.675	0.004	0.957	0.008	0.317	0.006	0.351
S3 sharpness regional	ds3	0.671	0.005	0.956	0.011	0.307	0.009	0.359
Laplace sharpness regional	ds1	0.678	0.005	0.963	0.009	0.32	0.01	0.102
Laplace sharpness regional	ds2	0.678	0.003	0.967	0.006	0.319	0.008	0.11
Laplace sharpness regional	ds3	0.676	0.004	0.964	0.012	0.317	0.007	0.113
BRISQUE	ds1	0.678	0.004	0.961	0.006	0.321	0.009	0.865
BRISQUE	ds2	0.678	0.004	0.96	0.012	0.321	0.009	0.983
BRISQUE	ds3	0.679	0.004	0.956	0.008	0.326	0.009	1.017
S index	ds1	0.679	0.004	0.958	0.008	0.325	0.008	3.788
S index	ds2	0.678	0.002	0.955	0.007	0.323	0.004	4.093
S index	ds3	0.677	0.003	0.959	0.008	0.32	0.008	4.180
Simple Gabor	ds1	0.669	0.004	0.972	0.011	0.297	0.007	8.587
Simple Gabor	ds2	0.665	0.004	0.971	0.008	0.289	0.008	9.327
Simple Gabor	ds3	0.666	0.003	0.974	0.009	0.291	0.007	9.533
Gradient Gabor	ds1	0.669	0.004	0.963	0.007	0.302	0.008	4.463
Gradient Gabor	ds2	0.665	0.005	0.957	0.012	0.293	0.012	4.848
Gradient Gabor	ds3	0.669	0.004	0.96	0.007	0.303	0.009	4.955
IL-NIQE colors	ds1	0.677	0.004	0.968	0.007	0.316	0.007	1.459
IL-NIQE colors	ds2	0.679	0.003	0.968	0.008	0.321	0.005	1.594
IL-NIQE colors	ds3	0.674	0.004	0.966	0.01	0.312	0.009	1.634
Gradient IL-NIQE	ds1	0.676	0.004	0.969	0.005	0.315	0.008	2.798
Gradient IL-NIQE	ds2	0.678	0.002	0.964	0.007	0.32	0.005	3.053
Gradient IL-NIQE	ds3	0.678	0.004	0.955	0.007	0.323	0.008	3.126

Downscaled features	Size	Accuracy		Precision neg.		Recall neg.		Time gain (sec)
		mean	std	mean	std	mean	std	
All features	ds1	0.646	0.003	0.936	0.011	0.257	0.005	24.625
All features	ds2	0.62	0.004	0.917	0.019	0.203	0.008	26.819
All features	ds3	0.63	0.003	0.912	0.012	0.228	0.007	27.443

Table A.3: Evaluation of the models with the threshold 0.2 and 38 features when extracting specified features from downscaled images.

Appendix B

Comparison of enhancement techniques combinations

Enhancement method	Worsenings			Improvements		
	From positive		From part.	From negative		From part.
	To part.	To neg.	To neg.	To part.	To pos.	To pos.
Gray-scaling	9	19	11	10	6	20
Binarization	34	26	35	17	15	28
CUM $\alpha = 0.2$, $\sigma^2 = 20$	15	23	28	18	10	29
DCT alpha rooting $\alpha = 0.975$	9	12	14	22	11	26
DFT UM $\sigma^2 = 50$	**8**	14	**9**	15	7	20
Laplace $\epsilon = 1/16$	9	11	16	15	7	23
CUM + Laplace	13	18	26	17	16	31
Laplace + CUM	18	14	25	11	**17**	**32**
CUM + Laplace + Binarize	36	22	39	16	14	29
Alpha rooting + Laplace	11	12	16	18	5	26
Laplace + Alpha rooting	**8**	**10**	14	21	8	26
Alpha rooting + Laplace + Binarize	68	107	46	18	7	16
Masking + Laplace	10	12	17	**23**	5	18
Laplace + Masking	10	12	18	21	3	20
Masking + Laplace + Binarize	33	25	35	18	14	24
PP full	24	21	18	22	10	25
CUM + PP full	21	17	21	15	16	27
CUM + PP full + Binarize	36	18	39	22	11	30
Masking + PP full	20	16	20	19	10	29
Masking + PP full + Binarize	33	31	39	20	12	29
Laplace + PP full	25	17	22	14	14	**32**
Laplace + PP full + Binarize	26	33	39	15	14	28

Table B.1: Comparison of several combinations of enhancement techniques and the same techniques used on their own

Enhancement method	Pos	Partial	Neg
Original	314	112	159
Gray-scaling	312	100	173
Binarization	297	100	188
CUM $\alpha = 0.2$, $\sigma^2 = 20$	315	88	182
DCT alpha rooting $\alpha = 0.975$	330	103	**152**
DFT UM $\sigma^2 = 50$	319	106	160
Laplace $\epsilon = 1/16$	324	97	164
CUM + Laplace	330	85	170
Laplace + CUM	**331**	84	170
CUM + Laplace + Binarize	299	96	190
Alpha rooting + Laplace	322	99	164
Laplace + Alpha rooting	330	101	154
Alpha rooting + Laplace + Binarize	162	136	287
Masking + Laplace	315	110	160
Laplace + Masking	315	105	165
Masking + Laplace + Binarize	294	104	187
PP full	304	115	166
CUM + PP full	319	100	166
CUM + PP full + Binarize	301	101	183
Masking + PP full	317	102	166
Masking + PP full + Binarize	291	97	197
Laplace + PP full	318	97	170
Laplace + PP full + Binarize	297	86	202

Table B.2: Comparison of total number of positive, partial and negative examples after application of several combinations of enhancement techniques and the same techniques used on their own

Bibliography

[1] Image text recognition apis showdown. `https://dataturks.com/blog/compare-image-text-recognition-apis.php`.

[2] Microsoft api documentation. `https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-recognizing-text#ocr-optical-character-recognition-api`.

[3] sklearn documentation. `https://scikit-learn.org/stable/`.

[4] Sos S Agaian, Karen Panetta, and Artyom M Grigoryan. Transform-based image enhancement algorithms with performance measure. *IEEE Transactions on image processing*, 10(3):367–382, 2001.

[5] Sabzali Aghagolzadeh and Okan K Ersoy. Transform image enhancement. *Optical Engineering*, 31(3):614–627, 1992.

[6] Sergey Bezryadin, Pavel Bourov, and Dmitry Ilinih. Brightness calculation in digital image processing. In *International symposium on technologies for digital photo fulfillment*, volume 2007, pages 10–15. Society for Imaging Science and Technology, 2007.

[7] Radu Ciprian Bilcu and Markku Vehvilainen. Constrained unsharp masking for image enhancement. In *International Conference on Image and Signal Processing*, pages 10–19. Springer, 2008.

[8] Gwendoline Blanchet and Lionel Moisan. An explicit sharpness index related to global phase coherence. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1065–1068. IEEE, 2012.

[9] Gwendoline Blanchet, Lionel Moisan, and Bernard Rougé. Measuring the global phase coherence of an image. In *2008 15th IEEE International Conference on Image Processing*, pages 1176–1179. IEEE, 2008.

[10] RH Britt. Optical character reader with skew correction. *Lundy Electronics and Systems, Inc., Charlotte, NC, US Patent*, 4(941):189, 1990.

[11] Gulcin Caner and Ismail Haritaoglu. Shape-dna: effective character restoration and enhancement for arabic text documents. In *2010 20th International Conference on Pattern Recognition*, pages 2053–2056. IEEE, 2010.

[12] Gang Cao, Lihui Huang, Huawei Tian, Xianglin Huang, Yongbin Wang, and Ruicong Zhi. Contrast enhancement of brightness-distorted images by improved adaptive gamma correction. *Computers & Electrical Engineering*, 66:569–582, 2018.

[13] Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, and Soumya K Ghosh. Optical character recognition systems. In *Optical Character Recognition Systems for Different Languages with Soft Computing*, pages 9–41. Springer, 2017.

[14] BB Chaudhuri and U Pal. Skew angle detection of digitized indian script documents. *IEEE Transactions on pattern analysis and machine intelligence*, 19(2):182–186, 1997.

[15] Bidyut Baran Chaudhuri, Pulak Kundu, and Nirupam Sarkar. Detection and gradation of oriented texture. 1993.

[16] Xiangrong Chen and Alan L Yuille. Detecting and reading text in natural scenes. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–II. IEEE, 2004.

[17] Yen-Lin Chen and Bing-Fei Wu. A multi-plane approach for text segmentation of complex document images. *Pattern Recognition*, 42(7):1419–1444, 2009.

[18] Yao-Yi Chiang and Craig A Knoblock. An approach for recognizing text labels in raster maps. In *2010 20th International Conference on Pattern Recognition*, pages 3199–3202. IEEE, 2010.

[19] Yao-Yi Chiang and Craig A Knoblock. Recognition of multi-oriented, multi-sized, and curved text. In *2011 International Conference on Document Analysis and Recognition*, pages 1399–1403. IEEE, 2011.

[20] Hojin Cho, Jue Wang, and Seungyong Lee. Text image deblurring using text-specific properties. In *European Conference on Computer Vision*, pages 524–537. Springer, 2012.

[21] Min Goo Choi, Jung Hoon Jung, and Jae Wook Jeon. No-reference image quality assessment using blur and noise. *International Journal of Computer Science and Engineering*, 3(2):76–80, 2009.

[22] Liu Chun-Lin. A tutorial of the wavelet transform. *NTUEE, Taiwan*, 2010.

[23] G Ciardiello, G Scafuro, MT Degrandi, MR Spada, and MP Roccotelli. An experimental system for office document handling and text recognition. In *Proc 9th Int. Conf. on Pattern Recognition*, pages 739–743, 1988.

[24] Shay B Cohen, Eytan Ruppin, and Gideon Dror. Feature selection based on the shapley value. In *IJCAI*, volume 5, pages 665–670, 2005.

[25] Amit Kumar Das and Bhabatosh Chanda. A fast algorithm for skew detection of document images using morphology. *International Journal on Document Analysis and Recognition*, 4(2):109–114, 2001.

[26] Ingrid Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE transactions on information theory*, 36(5):961–1005, 1990.

[27] Agnès Desolneux, Saïd Ladjal, Lionel Moisan, and J-M Morel. Dequantizing image orientation. *IEEE Transactions on Image Processing*, 11(10):1129–1140, 2002.

[28] Zeyang Dou, Kun Gao, Xiaodian Zhang, and Hong Wang. Fast blind image deblurring using smoothing-enhancing regularizer. *IEEE Access*, 7:90904–90915, 2019.

[29] Boris Epshtein, Eyal Ofek, and Yonatan Wexler. Detecting text in natural scenes with stroke width transform. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2963–2970. IEEE, 2010.

[30] Yuming Fang, Kede Ma, Zhou Wang, Weisi Lin, Zhijun Fang, and Guangtao Zhai. No-reference quality assessment of contrast-distorted images based on natural scene statistics. *IEEE Signal Processing Letters*, 22(7):838–842, 2014.

[31] Rony Ferzli and Lina J Karam. A no-reference objective image sharpness metric based on the notion of just noticeable blur (jnb). *IEEE transactions on image processing*, 18(4):717–728, 2009.

[32] Sylvain Fischer, Filip Šroubek, Laurent Perrinet, Rafael Redondo, and Gabriel Cristóbal. Self-invertible 2d log-gabor wavelets. *International Journal of Computer Vision*, 75(2):231–246, 2007.

[33] Salvador Gabarda and Gabriel Cristóbal. Blind image quality assessment through anisotropy. *JOSA A*, 24(12):B42–B51, 2007.

[34] Hayit Greenspan, Charles H Anderson, and Sofia Akber. Image enhancement by nonlinear extrapolation in frequency space. *IEEE Transactions on Image Processing*, 9(6):1035–1048, 2000.

[35] Ke Gu, Weisi Lin, Guangtao Zhai, Xiaokang Yang, Wenjun Zhang, and Chang Wen Chen. No-reference quality metric of contrast-distorted images based on information maximization. *IEEE transactions on cybernetics*, 47(12):4559–4565, 2016.

[36] Ke Gu, Dacheng Tao, Jun-Fei Qiao, and Weisi Lin. Learning a no-reference quality assessment model of enhanced images with big data. *IEEE transactions on neural networks and learning systems*, 29(4):1301–1313, 2017.

[37] Ke Gu, Guangtao Zhai, Weisi Lin, Xiaokang Yang, and Wenjun Zhang. Visual saliency detection with free energy theory. *IEEE Signal Processing Letters*, 22(10):1552–1555, 2015.

[38] Ke Gu, Guangtao Zhai, Xiaokang Yang, and Wenjun Zhang. Using free energy principle for blind image quality assessment. *IEEE Transactions on Multimedia*, 17(1):50–63, 2014.

[39] Zineb Hadjadj, Mohamed Cheriet, Abdelkrim Meziane, and Yazid Cherfa. A new efficient binarization method: application to degraded historical document images. *Signal, Image and Video Processing*, 11(6):1155–1162, 2017.

[40] Zineb Hadjadj, Abdelkrim Meziane, Yazid Cherfa, Mohamed Cheriet, and Insaf Setitra. Isauvola: Improved sauvola's algorithm for document image binarization. In *International Conference on Image Analysis and Recognition*, pages 737–745. Springer, 2016.

[41] Yu Han, Xiaoming Xu, and Yunze Cai. Novel no-reference image blur metric based on block-based discrete cosine transform statistics. *Optical Engineering*, 49(5):050501, 2010.

[42] Rania Hassen, Zhou Wang, and Magdy MA Salama. Image sharpness assessment based on local phase coherence. *IEEE Transactions on Image Processing*, 22(7):2798–2810, 2013.

[43] Stuart C Hinds, James L Fisher, and Donald P D'Amato. A document skew detection method using run-length encoding and the hough transform. In *[1990] Proceedings. 10th International Conference on Pattern Recognition*, volume 1, pages 464–468. IEEE, 1990.

[44] Christopher Hollitt and Ahmed Sheikh Deeb. Determining image orientation using the hough and fourier transforms. In *Proceedings of the 27th Conference on Image and Vision Computing New Zealand*, pages 346–351, 2012.

[45] Kai Huang, Zixuan Chen, Min Yu, Xiaolang Yan, and Aiguo Yin. An efficient document skew detection method using probability model and q test. *Electronics*, 9(1):55, 2020.

[46] John Immerkaer. Fast noise variance estimation. *Computer vision and image understanding*, 64(2):300–302, 1996.

[47] Anil K Jain. *Fundamentals of digital image processing.* Englewood Cliffs, NJ: Prentice Hall,, 1989.

[48] T-L Ji, Malur K Sundareshan, and Hans Roehrig. Adaptive image contrast enhancement based on human visual properties. *IEEE transactions on medical imaging*, 13(4):573–586, 1994.

[49] Christopher Kanan and Garrison W Cottrell. Color-to-grayscale: does the method matter in image recognition? *PloS one*, 7(1), 2012.

[50] Sang Ho Kim and Jan P Allebach. Optimal unsharp mask for image sharpening and noise removal. *Journal of Electronic Imaging*, 14(2):023005, 2005.

[51] Nick Kingsbury and Julian Magarey. Wavelet transforms in image processing. In *Signal analysis and prediction*, pages 27–46. Springer, 1998.

[52] Nick Kingsbury and Julian Magarey. *Wavelet Transforms in Image Processing*, pages 27–46. Birkhäuser Boston, Boston, MA, 1998.

[53] S-J Ko and Yong Hoon Lee. Center weighted median filters and their applications to image enhancement. *IEEE transactions on circuits and systems*, 38(9):984–993, 1991.

[54] Heljä Kukkonen, Jyrki Rovamo, Kaisa Tiippana, and Risto Näsänen. Michelson contrast, rms contrast and energy of various spatial stimuli at threshold. *Vision Research*, 33(10):1431–1436, 1993.

[55] Stephen W Lam et al. Vc: Skew detection using directional profile analysis. In *In: Proc. IAPR Workshop on Machine Vision and Application.* Citeseer, 1994.

[56] Anupama B Lamb and Madhuri Khambete. No-reference perceived image quality measurement for multiple distortions. *Multimedia Tools and Applications*, 77(7):8653–8675, 2018.

[57] Nour-Eddine Lasmar, Youssef Stitou, and Yannick Berthoumieu. Multiscale skewed heavy tailed model for texture analysis. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 2281–2284. IEEE, 2009.

[58] Daniel S Le, George R Thoma, and Harry Wechsler. Automated page orientation and skew angle detection for binary document images. *Pattern Recognition*, 27(10):1325–1344, 1994.

[59] Arthur Leclaire and Lionel Moisan. No-reference image quality assessment and blind deblurring with sharpness metrics exploiting fourier phase information. *Journal of Mathematical Imaging and Vision*, 52(1):145–172, 2015.

[60] Jong-Sen Lee. Digital image smoothing and the sigma filter. *Computer vision, graphics, and image processing*, 24(2):255–269, 1983.

[61] Xiaoyu Li, Bo Zhang, Jing Liao, and Pedro V Sander. Document rectification and illumination correction using a patch-based cnn. *ACM Transactions on Graphics (TOG)*, 38(6):1–11, 2019.

[62] Jian Liang, Daniel DeMenthon, and David Doermann. Geometric rectification of camera-captured document images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(4):591–605, 2008.

[63] Lixiong Liu, Bao Liu, Hua Huang, and Alan Conrad Bovik. No-reference image quality assessment based on spatial and spectral entropies. *Signal Processing: Image Communication*, 29(8):856–863, 2014.

[64] Xinhao Liu, Masayuki Tanaka, and Masatoshi Okutomi. Single-image noise level estimation for blind denoising. *IEEE transactions on image processing*, 22(12):5226–5237, 2013.

[65] Xinwei Liu, Marius Pedersen, Christophe M Charrier, and Patrick Bours. Performance evaluation of no-reference image quality metrics for face biometric images. *Journal of Electronic Imaging*, 27(2):023001, 2018.

[66] Kede Ma, Wentao Liu, Tongliang Liu, Zhou Wang, and Dacheng Tao. dipiq: Blind image quality assessment by learning-to-rank discriminable image pairs. *IEEE Transactions on Image Processing*, 26(8):3951–3964, 2017.

[67] Kede Ma, Wentao Liu, Kai Zhang, Zhengfang Duanmu, Zhou Wang, and Wangmeng Zuo. End-to-end blind image quality assessment using deep neural networks. *IEEE Transactions on Image Processing*, 27(3):1202–1213, 2017.

[68] Raman Maini and Himanshu Aggarwal. A comprehensive review of image enhancement techniques. *arXiv preprint arXiv:1003.4053*, 2010.

[69] Pina Marziliano, Frederic Dufaux, Stefan Winkler, and Touradj Ebrahimi. Perceptual blur and ringing metrics: application to jpeg2000. *Signal processing: Image communication*, 19(2):163–172, 2004.

[70] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[71] Lionel Moisan. Periodic plus smooth image decomposition. *Journal of Mathematical Imaging and Vision*, 39(2):161–179, 2011.

[72] Anush Krishna Moorthy and Alan Conrad Bovik. A two-step framework for constructing blind image quality indices. *IEEE Signal processing letters*, 17(5):513–516, 2010.

[73] Niranjan D Narvekar and Lina J Karam. A no-reference image blur metric based on the cumulative probability of blur detection (cpbd). *IEEE Transactions on Image Processing*, 20(9):2678–2683, 2011.

[74] Nikos Nikolaou and Nikos Papamarkos. Color reduction for complex document images. *International Journal of Imaging Systems and Technology*, 19(1):14–26, 2009.

[75] Olukayode A Ojo and Tatiana G Kwaaitaal-Spassova. An algorithm for integrated noise reduction and sharpness enhancement. *IEEE Transactions on Consumer Electronics*, 46(3):474–480, 2000.

[76] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[77] Stanley Osher and Leonid I Rudin. Feature-oriented image enhancement using shock filters. *SIAM Journal on numerical analysis*, 27(4):919–940, 1990.

[78] R Øyvind. Applications of the wavelet transform in image processing. *Department of informatics, University of Oslo (November 12, 2004)*.

[79] U Pal and BB Chaudhuri. An improved document skew angle estimation technique. 1996.

[80] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2908, 2014.

[81] José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martinez, and Joaquín Fernández-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 3, pages 314–317. IEEE, 2000.

[82] Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7):629–639, 1990.

[83] Cuong Cao Pham, Synh Viet Uyen Ha, and Jae Wook Jeon. Adaptive guided image filtering for sharpness enhancement and noise reduction. In *Pacific-Rim Symposium on Image and Video Technology*, pages 323–334. Springer, 2011.

[84] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[85] Wolfgang Postl. Detection of linear oblique structures and skew scan in digitized documents. In *Proc. Int. Conf. on Pattern Recognition*, pages 687–689, 1986.

[86] Tania Pouli, Douglas W Cunningham, and Erik Reinhard. A survey of image statistics relevant to computer graphics. In *Computer Graphics Forum*, volume 30, pages 1761–1788. Wiley Online Library, 2011.

[87] Shanto Rahman, Md Mostafijur Rahman, Mohammad Abdullah-Al-Wadud, Golam Dastegir Al-Quaderi, and Mohammad Shoyaib. An adaptive gamma correction for image enhancement. *EURASIP Journal on Image and Video Processing*, 2016(1):1–13, 2016.

[88] Zia-ur Rahman, Daniel J Jobson, and Glenn A Woodell. Retinex processing for automatic image enhancement. *Journal of Electronic imaging*, 13(1):100–111, 2004.

[89] Alevoor Ravishankar Rao. *A taxonomy for texture description and identification.* University of Michigan, 1989.

[90] Azriel Rosenfeld. *Digital picture processing.* Academic press, 1976.

[91] Michele A Saad, Alan C Bovik, and Christophe Charrier. Dct statistics model-based blind image quality assessment. In *2011 18th IEEE International Conference on Image Processing*, pages 3093–3096. IEEE, 2011.

[92] S Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.

[93] Panagiotis Saragiotis and Nikos Papamarkos. Local skew correction in documents. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(04):691–710, 2008.

[94] Jaakko Sauvola and Matti Pietikäinen. Skew angle detection using texture direction analysis. *image*, 100(1):T1, 1995.

[95] Hamid R Sheikh, Alan C Bovik, and Lawrence Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *IEEE Transactions on Image Processing*, 14(11):1918–1927, 2005.

[96] Ray Smith. An overview of the tesseract ocr engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 629–633. IEEE, 2007.

[97] Schuyler Smith. Subjective image quality measurement.

[98] Pranav Sodhani, Akshat Bordia, and Kannan Karthik. Blind content independent noise estimation for multimedia applications. *Procedia Computer Science*, 54:549–557, 2015.

[99] Changming Sun and Deyi Si. Skew and slant correction for document images using gradient direction. In *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, volume 1, pages 142–146. IEEE, 1997.

[100] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *European conference on computer vision*, pages 56–72. Springer, 2016.

[101] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 839–846. IEEE, 1998.

[102] Cuong T Vu and Damon M Chandler. S3: a spectral and spatial sharpness measure. In *2009 First International Conference on Advances in Multimedia*, pages 37–43. IEEE, 2009.

[103] Phong V Vu and Damon M Chandler. A fast wavelet-based algorithm for global and local image sharpness estimation. *IEEE Signal Processing Letters*, 19(7):423–426, 2012.

[104] David CC Wang, Anthony H Vagnucci, and CC Li. Digital image enhancement: a survey. *Computer Vision, Graphics, and Image Processing*, 24(3):363–381, 1983.

[105] Zhou Wang, Hamid R Sheikh, and Alan C Bovik. No-reference perceptual quality assessment of jpeg compressed images. In *Proceedings. International Conference on Image Processing*, volume 1, pages I–I. IEEE, 2002.

[106] Zhou Wang and Eero P Simoncelli. Local phase coherence and the perception of blur. In *Advances in neural information processing systems*, pages 1435–1442, 2004.

[107] Xiangyu Xu, Deqing Sun, Jinshan Pan, Yujin Zhang, Hanspeter Pfister, and Ming-Hsuan Yang. Learning to super-resolve blurry face and text images. In *Proceedings of the IEEE international conference on computer vision*, pages 251–260, 2017.

[108] Wufeng Xue, Lei Zhang, and Xuanqin Mou. Learning without human scores for blind image quality assessment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 995–1002, 2013.

[109] Hong Yan. Skew correction of document images using interline cross-correlation. *CVGIP: Graphical Models and Image Processing*, 55(6):538–543, 1993.

[110] Jia Yan, Jie Li, and Xin Fu. No-reference quality assessment of contrast-distorted images using contrast enhancement. *arXiv preprint arXiv:1904.08879*, 2019.

[111] Qixiang Ye and David Doermann. Text detection and recognition in imagery: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 37(7):1480–1500, 2014.

[112] Qixiang Ye, Qingming Huang, Wen Gao, and Debin Zhao. Fast and robust text detection in images and video frames. *Image and vision computing*, 23(6):565–576, 2005.

[113] Buyue Zhang and Jan P Allebach. Adaptive bilateral filter for sharpness enhancement and noise removal. *IEEE transactions on Image Processing*, 17(5):664–678, 2008.

[114] Lin Zhang, Lei Zhang, and Alan C Bovik. A feature-enriched completely blind image quality evaluator. *IEEE Transactions on Image Processing*, 24(8):2579–2591, 2015.

[115] Xu Zhao, Kai-Hsiang Lin, Yun Fu, Yuxiao Hu, Yuncai Liu, and Thomas S Huang. Text from corners: a novel approach to detect text and caption in videos. *IEEE Transactions on Image Processing*, 20(3):790–799, 2010.

[116] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017.

[117] Tong Zhu and Lina Karam. A no-reference objective image quality metric based on perceptually weighted local noise. *EURASIP Journal on Image and Video Processing*, 2014(1):5, 2014.

[118] Xiang Zhu and Peyman Milanfar. A no-reference sharpness metric sensitive to blur and noise. In *2009 International Workshop on Quality of Multimedia Experience*, pages 64–69. IEEE, 2009.

[119] Xiang Zhu and Peyman Milanfar. Automatic parameter selection for denoising algorithms using a no-reference measure of image content. *IEEE transactions on image processing*, 19(12):3116–3132, 2010.