Friedrich-Alexander-Universität Erlangen-Nürnberg

Lehrstuhl für Multimediakommunikation und Signalverarbeitung

Prof. Dr.-Ing. André Kaup

Master Thesis

Text Recognition Algorithms for Screen Content Quality Assessment

Sebastian Hirt

July 2023

Supervisors: Prof. Dr.-Ing. André Kaup
M. Sc. Hannah Och

Lehrstuhl für Multimediakommunikation
und Signalverarbeitung
Prof. Dr.-Ing. André Kaup

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Master Thesis

for

Mr. Sebastian Hirt

Text Recognition Algorithms for Screen Content Quality Assessment

Texterkennungsalgorithmen für die Qualitätsbewertung von Bildschirminhalten

Screen content (SC) refers to images and videos as they can be found on screens during office work or similar. Consequently, next to buttons, icons, and computer graphics, they contain textual information in most cases. For images containing text, human viewers place a high importance on readability of text when assessing image quality. However, conventional objective image quality assessment algorithms, such as PSNR, MS-SSIM or even screen content specific quality metrics, such as ESIM or GFM, only consider text readability indirectly, e.g. by taking into account gradient distortions, or not at all.

Since text readability plays an important role in the subjective quality assessment of screen content images, evaluating text recognition rates can be a useful addition to conventional quality metrics. However, subjective tests are not feasible, since they are too expensive and time-consuming. Instead, current Deep Learning-based text detection and recognition algorithms have shown very high text recognition rates and can be utilized to simulate the human reader.

In this thesis, Mr. Hirt will explore the application of such algorithms for the assessment of screen content image quality. First, Mr. Hirt will research state-of-the-art text recognition and detection methods. Since ground truth textual information is not included in available screen content datasets, Mr. Hirt will generate a labeled dataset to evaluate the efficiency of the researched algorithms on screen content data. Available datasets with subjective quality scores will be utilized to investigate the correlation between text recognition rates and human judgement. Since most datasets do not contain textual ground truth information, in a further step, Mr Hirt will investigate the feasibility of using recognized text from pristine images as ground truth instead. A structured implementation and detailed documentation of the framework and the performed experiments is part of the work.

Start: 01.02.2023
End: 31.07.2023

A. Kaup

Prof. Dr.-Ing. A. Kaup

Declaration

I confirm that I have written this thesis unaided and without using sources other than those listed and that this thesis has never been submitted to another examination authority and accepted as part of an examination achievement, neither in this form nor in a similar form. All content that was taken from a third party either verbatim or in substance has been acknowledged as such.

Place, Date	Sebastian Hirt Rennfeld 5b, 91792 Ellingen

Contents

Kurzfassung

In der heutigen digital vernetzten Welt spielen Bildschirminhalte in verschiedenen Anwendungen wie Videokonferenzen, Remote-Desktop-Zugriff und Videostreaming eine wichtige Rolle, so dass die Bildqualität ein entscheidender Aspekt für die Verbesserung der Benutzerfreundlichkeit ist. Herkömmliche Methoden zur Bewertung der Bildqualität, wie die peak signal-to-noise ratio (PSNR) und der structural similarity index (SSIM), sind jedoch für Bildschirminhalte mit Text unzureichend. In dieser Arbeit wird die Anwendung von optical character recognition (OCR) Algorithmen zur Bewertung der Bildqualität von Bildschirminhalten untersucht. Zunächst untersuchen wir den Stand der Technik von OCR Methoden und vergleichen die Leistung von Tesseract OCR und EasyOCR anhand des SCID-Datensatzes. Da der Datensatz keine Textbeschriftungen enthält, annotieren wir den Datensatz, um die Effektivität der optischen Zeichenerkennungsmethoden zu bewerten. Außerdem untersuchen wir die Korrelation zwischen der Leistung der OCR Algorithmen und dem menschlichen Urteilsvermögen an Bildern mit verschiedenen Arten von Verzerrungen, indem wir die character error rate (CER) mit der in dem Datensatz enthaltenen subjektiven mean opinion scores (MOSs) vergleichen. Darüber hinaus erweitern wir den SCID-Datensatz mit Bildern, die mit high efficiency video coding (HEVC) und versatile video coding (VVC) Codecs verzerrt sind. Diese Erweiterung ermöglicht es uns zu untersuchen, ob OCR Algorithmen als zuverlässige Basiswahrheit für den Vergleich verschiedener Codecs dienen kann, indem wir die Bjøntegaard-Deltaraten zwischen verschiedenen Raten-Verzerrungs Kurven berechnen. Unsere Ergebnisse deuten darauf hin, dass die OCR Algorithmen vielversprechende Werkzeuge für die Bewertung der Bildqualität von Bildschirminhalten bei bestimmten Arten von Verzerrungen sind, insbesondere wenn sie durch andere Metriken, die graphische Bildinhalte bewerten, ergänzt werden. Zusätzlich legen unsere Ergebnisse nahe, dass sich EasyOCR als pseudo Grundwahrheit für den Vergleich von Codecs eignet.

Abstract

In today's digitally interconnected world, screen content images play a significant role in various applications such as video conferencing, remote desktop access, and video streaming, making image quality a crucial aspect for enhancing user experience. However, conventional image quality assessment methods like peak signal-to-noise ratio and structural similarity index are inadequate for screen content images containing text. This thesis explores the application of optical character recognition (OCR) algorithms for evaluating screen content image quality. Initially, we research state-of-the-art OCR methods, comparing the performance of Tesseract OCR and EasyOCR using the SCID dataset. As the dataset lacks text labels, we annotate the dataset to evaluate the effectiveness of the OCR methods. Additionally, we investigate the correlation between the performance of OCR and human judgment on images with different types of distortion by comparing the character error rate with the subjective mean opinion score included in the dataset. Furthermore, we expand the SCID dataset by incorporating images distorted with high-efficiency video coding and versatile video coding. This extension enables us to explore whether OCR can serve as a reliable ground truth for comparing different codecs, using the Bjøntegaard Delta Rates between different rate-distortion curves. Our findings suggest that OCR holds promise as a valuable tool for screen content image quality assessment for certain types of distortion, particularly when complemented with other metrics, that evaluate the quality of the graphical parts of images. Additionally, our results indicate that EasyOCR proves to be a suitable source for generating the pseudo ground truth for codec comparison.

Acronyms

BDRate Bjøntegaard delta rate

CC contrast change
CER character error rate
CNN convolutional neural network
CQD color quantization dither
CSC color saturation change
CTC connectionist temporal classification

ESIM edge similarity index

FR full-reference
FSIM feature similarity index

GB Gaussian blur
GFM Gabor feature-based model
GN Gaussian noise
GT ground truth

HEVC high efficiency video coding

IoU intersection over union
IQA image quality assessment

JPEG Joint Photographic Experts Group

LSTM long short-term memory

MB motion blur
MOS mean opinion score
MSE mean squared error

NR no-reference

OCR optical character recognition

PLCC Pearson linear correlation coefficient
PSNR peak signal-to-noise ratio

QP quantization parameter

RMSE root mean squared error
RR reduced-reference

SCC screen content coding
SCI screen content image
SCID screen content image database
SRCC Spearman rank correlation coefficient
SSIM structural similarity index

VQEG Video Quality Experts Group
VVC versatile video coding

Chapter 1

Introduction

In today's digital age, screen content plays a vital role in our daily lives. From office work to entertainment, we are constantly interacting with images and videos on screens. Many of these images contain text, graphics and user interface elements that are not found in natural images. As such, the quality of screen content is of utmost importance for the viewer. One key aspect of screen content quality is the readability of text. However, conventional objective image quality assessment (IQA) algorithms do not directly consider text readability. This is where optical character recognition (OCR) algorithms come into play.

In this thesis, we will explore the application of OCR algorithms for the assessment of screen content image quality. We research state-of-the-art OCR methods, generate the labels for a dataset and investigate the correlation between the performance of OCR and human judgment on images with different types of distortion. Additionally, we explore the feasibility of using the quality of OCR algorithms to compare codecs. Through a structured implementation and detailed documentation of the experiments, we provide valuable insights into the potential of OCR algorithms for screen content IQA. In this thesis, I use the pronouns *we* and *our* to refer to myself and the larger scientific community.

This thesis is structured as follows. In chapter 2, we give an overview of conventional IQA methods for natural images and adapted methods, that are used to assess screen content images. Further, we detail the evaluation procedure we apply to compare the performance of IQA methods. In chapter 3, we initially describe conventional OCR methods and then focus on the state-of-the-art OCR methods. Additionally, we introduce the two OCR methods, Tesseract OCR [1] and EasyOCR [2], which we use in our experiments. In chapter 4, we detail the different types of distortion in the SCID dataset [3] used in this thesis and describe the labeling procedure we employ to generate text labels for the dataset. Further, the extension of the dataset by incorporating images distorted with high efficiency video coding (HEVC) [4] and versatile video coding (VVC) [5] is explained. In chapter 5, we compare the performance of Tesseract

OCR and EasyOCR for the different types of distortion and investigate the correlation between the performance of OCR and human judgment. Afterwards, we explore the feasibility of using the predictions of the OCR algorithms to compare codecs. Finally, we summarize and conclude our findings in chapter 6.

Chapter 2

Quality Assessment

In this chapter, we begin by examining related work in the field of IQA for natural images. Afterwards, we describe the differences to screen content images (SCIs) and discuss some metrics better suited to evaluate the quality of SCIs. Finally, we review the specific procedure used in our thesis to evaluate the performance of IQA algorithms.

2.1 Conventional Quality Assessment

IQA is a research field dedicated to quantifying the quality of distorted images [6]. The term "quality" always refers to the quality of an image as perceived by the human visual system. However, there is an exception when images are used for machine learning or other tasks that do not involve human perception. In this case, image quality may be defined by the performance of the machine learning algorithm. This aspect might become important when determining the appropriate compression techniques for datasets used in machine learning or identifying the types of distortions that have the most significant impact on OCR algorithms. Consider, for instance, an application where an OCR algorithm is utilized to recognize text in a presentation, and the recognized text is then read aloud by a text-to-speech system for a blind person. In this scenario, the image quality is defined by the performance of the OCR algorithm rather than human perception.

Generally, we can divide IQA algorithms into three categories [6]: full-reference (FR), reduced-reference (RR) and no-reference (NR). The goal of FR algorithms is to predict the quality of a distorted image by comparing it to the same image without distortion, called the reference image. RR algorithms predict the quality of the distorted image by comparing a reduced number of features of the distorted image to the reference image. NR algorithms only use the distorted image to predict its quality directly. In this thesis, we focus on FR algorithms, as we have access to the original images. Another distinction can be made between the type of images that are assessed. We can differentiate between natural images and SCIs. Natural images can

be pictures of landscapes, people or objects and are captured by image sensors while SCIs are images of screen content, containing text, graphics or UI elements and are directly recorded in a digital format.

For natural images, one common metric is the peak signal-to-noise ratio (PSNR) [7]. It describes the ratio of the maximum possible power of a signal and the power of corrupting noise that affects it. When it is applied to an image, the PSNR is defined as

$$\mathrm{PSNR} = 10 \cdot \log_{10} \left(\frac{R^2}{\mathrm{MSE}} \right), \tag{2.1}$$

with R denoting the maximum possible pixel value of the image and the MSE describing the mean squared error (MSE) between the distorted and the reference image. Another common metric is the structural similarity index (SSIM) [8], which takes salient features of the images into account. It combines the luminance, the contrast and the structural differences between two images into one metric. Compared to the PSNR, the SSIM is more closely correlated with human perception of the image quality [9]. Additionally, the SSIM was extended to the MS-SSIM [10], which incorporates details at multiple scales of the image. This provides better coverage of the human visual system compared to the single scale approach, as it takes into account different viewing conditions and resolutions of the images. However, the SSIM and MS-SSIM are surpassed in all criteria by the feature similarity index (FSIM) [11]. The FSIM is based on two features, namely the phase congruency and the gradient magnitude. Phase congruency is used to identify the importance of local structures in the image, while the gradient magnitude is used to measure the local contrast and in general the local rate of change of pixel values. These two features are then combined into a single metric to assess the quality of the image.

2.2 Screen Content Specific Quality Assessment

SCIs are images directly recorded in a digital format, containing text, graphics and user interface elements. A neighboring field is document image quality assessment, which assesses the quality of scanned documents and is thus mainly concerned with text [12]. SCIs often contain a combination of document and natural image content. Therefore, they differ statistically from natural and document images. They often contain large areas of uniform color, sharp contrasts and geometric structures. Conventional IQA algorithms designed for natural images do not perform well in terms of correlation with the human perception of the image quality [13]. Thus, different metrics are required

to evaluate their the quality of SCIs. To address this issue, some research has been conducted.

In [14], researchers use a convolutional neural network to assess the quality of documents. This enables the automatic assessment of document quality to filter out low-quality documents, on which OCR algorithms would perform poorly or select high quality frames from a video recording of a document. The documents are segmented into text and non-text regions. Then, the proposed convolutional neural network (CNN) is applied to each of the text content patches to predict quality scores, which are averaged over the whole image. Finally, the resulting scores are analyzed for correlation with OCR performance. The CNN achieves state-of-the-art performance in assessing the quality of the documents. In [15], the authors propose an objective IQA metric for SCIs that considers the text and pictorial content of an image separately. For the pictorial regions, the luminance and structural features are extracted. On the other hand, for the regions containing text, the authors use the gradient information to predict the visual quality. Afterwards, the two scores are weighted and combined. The proposed method is called SFUW and shows superior performance compared to other screen content IQA metrics. Yang et al. [12] investigate a subjective quality score, that considers text regions, pictorial regions and the entire image separately. The authors find that the textual regions contribute more to the overall subjective quality of an image compared to the pictorial regions. Additionally, an objective IQA method is proposed that uses the weighting of the subjective scores to combine objective metrics that consider the different regions into one. Further, in [3], a new screen content IQA metric called edge similarity index (ESIM) is proposed. Compared to the other screen content metrics, ESIM does not consider different regions of the image separately. Instead, it calculates the edge contrast, the edge width and the edge direction of the whole image. Those three features are then compared between the distorted and the reference image. The resulting similarities are then combined into a single score by a pooling strategy, that computes the weighted average of the three metrics with the maximum edge width of the two images as its weighting factor. Despite the missing separation of the image into different regions, ESIM achieves state-of-the-art performance on the dataset used in our thesis compared to other metrics. Similarly, in [16], the authors propose a metric that uses the Gabor filter to extract features from the image, named Gabor feature-based model (GFM). The motivation to use the Gabor filter is that it yields edge information, which is highly consistent with the human visual system. The method compares the similarities of the two chrominance components of the distorted and the reference image. Additionally, the Gabor filters are applied to

the luminance component of both images and the similarities are compared. Finally, the two generated similarity maps are combined by the proposed Gabor-feature-based pooling strategy. This is done similarly to the pooling strategy of ESIM. Due to the similarity of the Gabor filter to parts of the receptive field of the human visual system, the authors used the maximum values of the two Gabor feature maps to weight the final quality metric. The GFM not only achieves superior performance compared to other screen content and natural image IQA metrics, but is also less computationally complex, especially compared to the ESIM. Compared to these metrics, the OCR methods only consider the text regions to assess the quality of the images.

2.3 Evaluation Procedure of Quality Assessment Algorithms

To evaluate the suitability of an objective IQA metric, in our case the character error rate (CER), more specifically the $\mathrm{CER_c}$, see section 3.5, and its correlation to the human subjective score, in this thesis the mean opinion score (MOS), see chapter 4, there are three aspects to consider [17][6], namely prediction consistency, prediction monotonicity, and prediction accuracy, which we describe in this section. Since the Video Quality Experts Group (VQEG) recommends removing nonlinearities from the data before calculating these metrics [17], we describe this procedure in the next subsection. In the following, in place of a generic quality metric, we use the $\mathrm{CER_c}$ as used in the remaining parts of this thesis.

2.3.1 Nonlinear Transformation

To remove nonlinearities, we fit a model to the $\mathrm{CER_c}$ and the MOS values. This model is described in [18][19]. Given the i-th image in our dataset, its MOS value is denoted as MOS_i, its $\mathrm{CER_c}$ value as $\mathrm{CER}_{\mathrm{c},i}$ and its predicted MOS value as $\mathrm{MOS}_{\mathrm{p},i}$. The corresponding vectors are defined as

$$\mathbf{CER}_\mathrm{c} = \begin{pmatrix} \mathrm{CER}_{\mathrm{c},1} \\ \mathrm{CER}_{\mathrm{c},2} \\ \vdots \\ \mathrm{CER}_{\mathrm{c},N} \end{pmatrix}, \ \mathbf{MOS} = \begin{pmatrix} \mathrm{MOS}_1 \\ \mathrm{MOS}_2 \\ \vdots \\ \mathrm{MOS}_N \end{pmatrix} \text{ and } \mathbf{MOS}_\mathrm{p} = \begin{pmatrix} \mathrm{MOS}_{\mathrm{p},1} \\ \mathrm{MOS}_{\mathrm{p},2} \\ \vdots \\ \mathrm{MOS}_{\mathrm{p},N} \end{pmatrix}, \tag{2.2}$$

with N being the number of images in the dataset. This number varies, depending on the experiment, as we are selecting a subset of the dataset for each experiment. We describe this selection in chapter 4 and declare which subset we use in each experiment.

Then, the model can be defined as

$$\mathbf{MOS}_\mathrm{p} = \frac{\beta_1 - \beta_2}{1 + \mathrm{e}^{-\left(\frac{\mathbf{CER}_\mathrm{c} - \beta_3}{|\beta_4|}\right)}} + \beta_2, \tag{2.3}$$

with β_1, β_2, β_3, and β_4 denoting the parameters of the model. Although the model in [17] is more recent, we could not find initial parameters for it and decided to work with the older model. Additionally, there are other, more recent, publications [3, 19, 20, 21, 11, 22, 23, 24, 25] that use the model proposed in [26], which does not specify initial conditions either. In [27], a solution to estimating the initial parameters is proposed, which is out of scope for this thesis. The parameters are initialized as

$$\begin{aligned} \beta_1 &= \max \mathbf{CER}_\mathrm{c} \\ \beta_2 &= \min \mathbf{CER}_\mathrm{c} \\ \beta_3 &= \overline{\mathrm{MOS}} \\ \beta_4 &= 1. \end{aligned} \tag{2.4}$$

The parameters are adjusted with the least squared method [28] until the model fits the data of all the images. The model and the $\mathrm{CER_c}$ values are then used to calculate the predicted $\mathrm{MOS_p}$ values.

In Figure 2.1, an example of the nonlinear fitting is depicted. The initial data consists of some randomly generated MOS and $\mathrm{CER_c}$ values. We can see, that the model with the initial parameters does not fit the data. The parameters are then adjusted by the least squares method [28] to fit the model to the data. Finally, we can observe that the fitted curve clearly fits the data better than the curve with the initial parameters. Now, the $\mathrm{MOS_p}$ values can be calculated by using the fitted model and the $\mathrm{CER_c}$ values. With the $\mathrm{MOS_p}$ the following three metrics [29] can be calculated.

2.3.2 Pearson Correlation

The Pearson linear correlation coefficient (PLCC) [30] describes the linear correlation between two variables, normalized to the range $[-1, 1]$. This metric is used to measure the prediction linearity and consistency of the method. It is defined as

$$\mathrm{PLCC} = \frac{\sum_{i=1}^{N} (\mathrm{MOS}_i - \overline{\mathrm{MOS}})(\mathrm{MOS}_{\mathrm{p},i} - \overline{\mathrm{MOS}}_\mathrm{p})}{\sqrt{\sum_{i=1}^{N} (\mathrm{MOS}_i - \overline{\mathrm{MOS}})^2 \sum_{i=1}^{N} (\mathrm{MOS}_{\mathrm{p},i} - \overline{\mathrm{MOS}}_\mathrm{p})^2}}, \tag{2.5}$$

with $\overline{\mathrm{MOS}}$ and $\overline{\mathrm{MOS}}_\mathrm{p}$ representing the mean values of the $\mathbf{MOS}$ and $\mathbf{MOS}_\mathrm{p}$ vectors

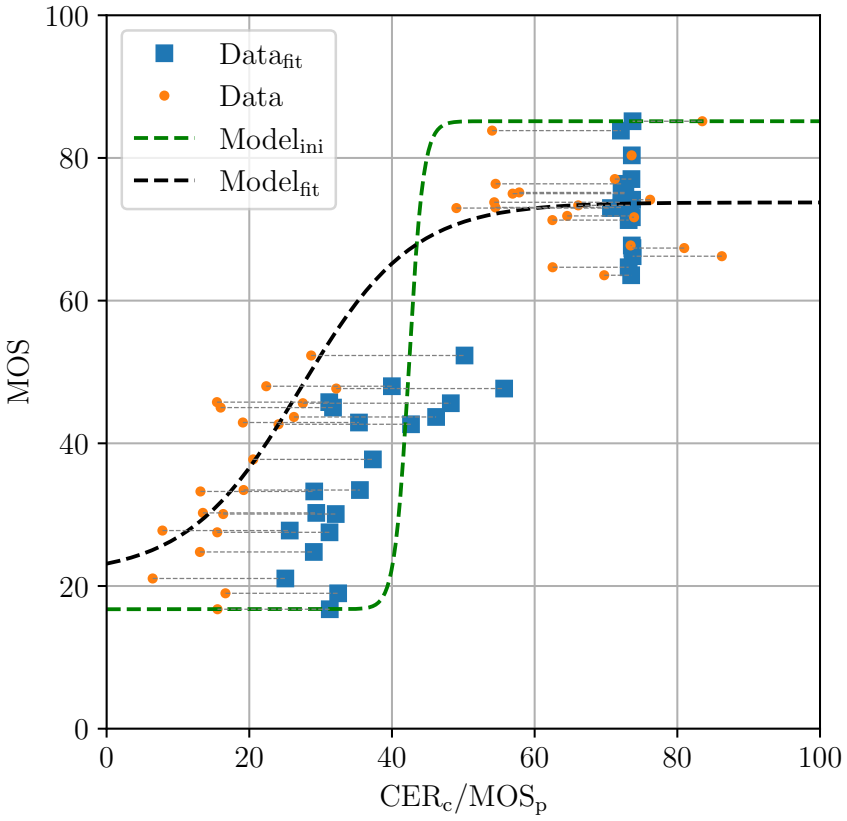


Figure 2.1: Example of the nonlinear fitting. Data before fitting (CER_c vs. MOS) and after fitting (MOS_p vs MOS). Model with initial parameters and fitted parameters.

respectively and N representing the total number of images in the dataset. If the PLCC is close to 1, the two vectors have a positive linear relationship, which means that if MOS_i increases, $\text{MOS}_{\text{p},i}$ increases as well. If the PLCC is close to -1, the two vectors have a negative linear relationship, which means that if MOS_i increases, $\text{MOS}_{\text{p},i}$ decreases. If the PLCC is close to 0, the two vectors have no correlation.

2.3.3 Spearman Ranked Correlation

The Spearman rank correlation coefficient (SRCC) [30] describes the monotonic correlation between two variables, normalized to the range $[-1, 1]$. Thus it is used to measure the prediction monotonicity of the method. Compared to the PLCC, it takes the rank, or order, of the values into account, not the exact values. The scores $\text{CER}_{\text{c},i}$ and MOS_i are transformed into their ranks $\text{CER}_{\text{c,r},i}$ and $\text{MOS}_{\text{r},i}$ respectively, with values in the range $[1, N]$. If for example, the first two values are tied, their rank is set to

the mean, in this case $(1+2)/2 = 1.5$. Note, that the fitting procedure does not matter for the SRCC, because the ranks between the $\mathrm{CER_c}$ and MOS values stay the same compared to the $\mathrm{MOS_p}$ and MOS values, because the fitted models are monotonic. With these values, the SRCC is defined as

$$\mathrm{SRCC} = \frac{\sum_{i=1}^{N} (\mathrm{MOS_{r,}}_i - \overline{\mathrm{MOS}}_\mathrm{r})(\mathrm{CER_{c,r,}}_i - \overline{\mathrm{CER}}_\mathrm{c,r})}{\sqrt{\sum_{i=1}^{N} (\mathrm{MOS_{r,}}_i - \overline{\mathrm{MOS}}_\mathrm{r})^2 \sum_{i=1}^{N} (\mathrm{CER_{c,r,}}_i - \overline{\mathrm{CER}}_\mathrm{c,r})^2}}, \tag{2.6}$$

with $\overline{\mathrm{MOS}}_\mathrm{r}$ and $\overline{\mathrm{CER}}_\mathrm{c,r}$ representing the mean values of the $\mathbf{MOS}_\mathrm{r}$ and $\mathbf{CER}_\mathrm{c,r}$ vectors respectively. If the SRCC is close to 1, the two vectors have a positive monotonic relationship, which means that the rank of the $\mathrm{CER_{c,}}_i$ increases, while the rank of the MOS_i increases. If the SRCC is close to -1, the two vectors have a negative monotonic relationship, which means that the rank of the $\mathrm{CER_{c,}}_i$ increases, while the rank of the MOS_i decreases. If the SRCC is close to 0, the ranks of the two vectors have no correlation. These characteristics help us to determine if the $\mathrm{CER_c}$ is a good predictor for the MOS, by investigating how similar the ranks of the two metrics are.

2.3.4 Root Mean Squared Error

The root mean squared error (RMSE) is a metric that measures the average magnitude of the error between the predicted values and the actual values. It is used to measure the prediction accuracy of the method. In our case it is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\mathrm{MOS_{p,}}_i - \mathrm{MOS}_i)^2}. \tag{2.7}$$

To summarize, PLCC measures the prediction linearity and consistency, SRCC measures the prediction monotonicity and the RMSE measures the prediction accuracy. With these, we can now determine if the $\mathrm{CER_c}$ is a good predictor for the MOS. It is a better predictor the larger the PLCC and SRCC values are and the smaller the RMSE is.

2.4 Bjøntegaard Delta Rate

The Bjøntegaard delta rate (BDRate) [31][32] is defined as the average difference between two rate-distortion curves of two codecs. The first curve is the reference curve and the second the test curve. Those curves are defined by a set of points $(R_\mathrm{k,i}, M_\mathrm{k,i})$, where $R_\mathrm{k,i}$ is the bitrate and $M_\mathrm{k,i}$ is the metric, in our case the $\mathrm{CER_c}$, of the image

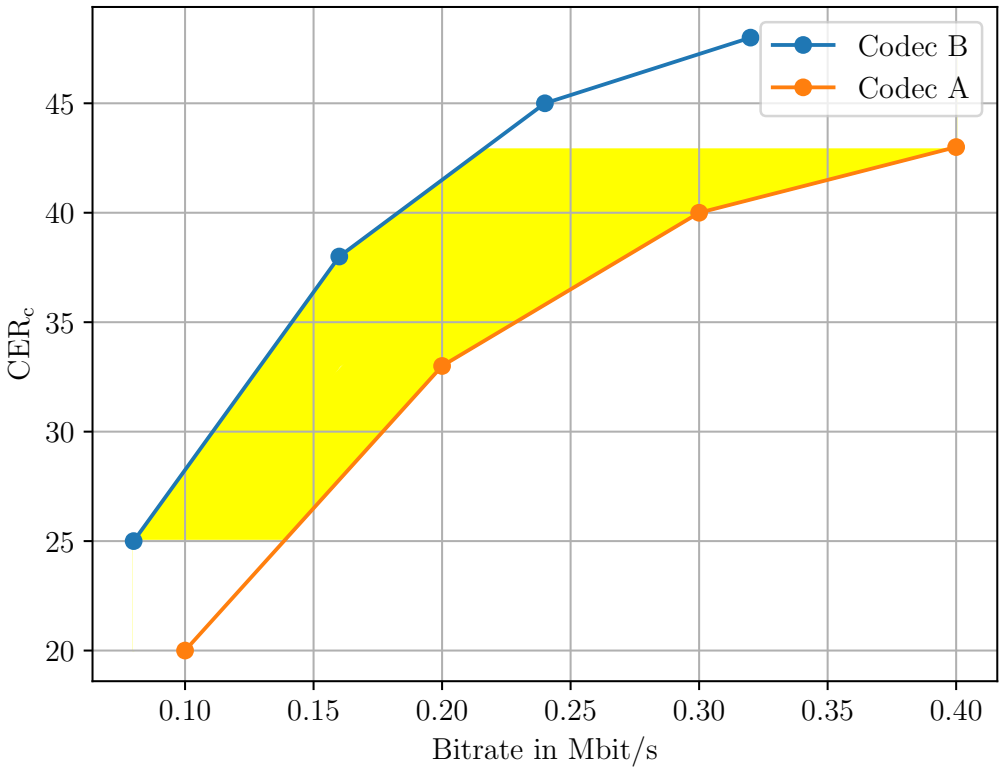


Figure 2.2: Example of the BDRate calculation with dummy values. Adapted from [32]

compressed by a codec k, with $k \in \{A, B\}$, with quantization parameter (QP) i, with $i \in \{35, 40, 45, 50\}$. The $R_{k,i}$ values are first converted to the logarithmic scale to not bias the results towards the higher bitrates with

$$r_{k,i} = \log_{10}(R_{k,i}) . \tag{2.8}$$

Those values in combination with the corresponding $M_{k,i}$ values are then used as anchor points for an interpolation with a third order polynomial. For our work, we use the Akima interpolation suggested in [32], which seems to show more accurate interpolation curves for a variety of metrics. The resulting functions are denoted by $\hat{r}_k$, respectively. The interpolation results in two curves, one for each codec, that pass through all anchor points. Finally, the BDRate can be denoted as ΔR and is calculated

by the integral of the difference between the two curves as

$$\Delta R = 10^{\frac{1}{M_{\text{low}} - M_{\text{high}}} \int_{M_{\text{low}}}^{M_{\text{high}}} \hat{r}_{\text{B}}(M) - \hat{r}_{\text{A}}(M) \mathrm{d}M} - 1. \tag{2.9}$$

The lower and upper bound of the integral are given by

$$\begin{aligned} M_{\text{low}} &= \max\left(M_{\text{A},50}, M_{\text{B},50}\right) \\ M_{\text{high}} &= \min\left(M_{\text{A},35}, M_{\text{B},35}\right). \end{aligned} \tag{2.10}$$

The bounds are the maximum of the lowest quality points and the minimum of the highest quality points and can be seen in Figure 2.2.

ΔR describes the average difference between the two curves in percent. This enables us to compare the rate-distortion curves of two codecs. The CER_c is a metric that describes the difference between two texts and is explored in more detail in section 3.5. In our case, we can calculate the CER_c for each codec in two ways: once with respect to the hand annotated text label, which we explain in section 4.2, and once in relation to the prediction of the OCR algorithms on the reference image without distortion. Throughout this thesis, we refer to the hand annotated text label as the true ground truth (GT) and the prediction of the OCR methods on the reference image as the pseudo GT. Thus, we can calculate the ΔR value for two codecs with respect to the true GT and the pseudo GT. These values can then be compared to evaluate, if using the OCR algorithms as a pseudo GT is a good estimation of the difference between codecs. If the difference between the two values is small, then the estimation is good and we might be able to use the OCR algorithms as a reference for future codec comparisons. We go into more detail about the specific codecs we use for our comparisons in section 4.4.

In this chapter, we have outlined the evaluation procedure for the CER_c values predicted by the OCR algorithms. However, to calculate the CER_c values, we need OCR algorithms to extract text from the images. Thus, in the following chapter we introduce the OCR algorithms that we use in our experiments.

Chapter 3

Optical Character Recognition

In this chapter, we summarize existing OCR methods and describe the two methods we use in this thesis, EasyOCR and Tesseract OCR, in detail. We also introduce the CER metric, which is used to evaluate the performance of the OCR methods.

3.1 Conventional Optical Character Recognition

OCR [33] generally involves the following steps to extract text from an image. First, the image is preprocessed, which might include binarization, noise removal, and skew correction. This step tries to improve the quality of the image from the perspective of the OCR method. Second, the image is segmented into individual characters, words or lines. Next, the segmented elements are categorized by Bayesian, nearest neighbor or neural network based classifiers. Finally, the recognized elements are postprocessed, which might include the use of multiple classifiers simultaneously and comparison of the results, incorporation of the context of the image or dictionary data to correct errors. Over the years, many different methods have been proposed to classify the characters based on different features [34][35]. The main goal was to find features, that are distinctive enough for all the relevant characters to achieve a high classification performance. These features might include bars, loops or hooks. Applications of OCR systems include handwriting recognition, receipt imaging, check processing in the banking industry and improvement of CAPTCHA systems, by regenerating the CAPTCHA until it cannot be recognized by the OCR method.

3.2 Neural Network Based Optical Character Recognition

In recent years, classifiers based on neural network based methods [36] have become more popular. Especially long short-term memory (LSTM) networks are prominent, because of their leverage of the context from previous characters or regions of the image. In general, neural network based methods perform better than conventional methods.

One of the most popular examples is Calamari [37], which is focused on recognizing text in historical documents. It uses the connectionist temporal classification (CTC) method to train a model composed of LSTM and CNN layers. Additionally, it includes the capability to employ multiple models simultaneously and use them to vote on the correct prediction. To make the voting meaningful, the models can be trained with different training datasets, have different model architectures or use different base models for finetuning. This enables a more robust final prediction and state-of-the-art performance on historical document datasets. Another example is the Inception V3 network [38], which implements a CNN to recognize printed text in images with poor quality. The authors leveraged fine tuning on a pre-trained base model to reduce the training time and improve the performance. The deep learning method shows significant improvements over traditional OCR methods, especially for low quality images. The OCR methods used in this thesis, EasyOCR and Tesseract OCR, are detailed in the following sections.

3.3 EasyOCR

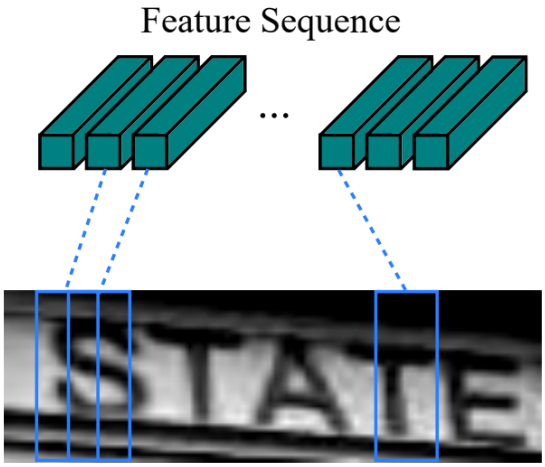


Figure 3.1: EasyOCR feature sequencing for an image of the word `state`, from [39].

EasyOCR is an open source Python library for OCR [2]. It leverages the CRAFT algorithm [40] for text detection, utilizing a fully convolutional network structure to predict word or character boxes within an image. For text recognition, EasyOCR adopts the None-VGG-BiLSTM-CTC architecture [39]. The model consists of CNN layers to extract relevant features from the input image. These features are then transformed into sequences, as illustrated in Figure 3.1. Then, a deep bidirectional LSTM

network is used to predict a character per frame of the feature sequence. This process generates output that may resemble something like `-ss-t-a-tt-e-`. The benefit of using the LSTM lies in its ability to incorporate contextual information from earlier frames of the sequence to enhance the recognition accuracy. Finally, the transcription layer uses CTC to combine the per-frame predictions into a single text prediction. For the example above, it does so by first removing the duplicate characters directly following each other, resulting in `-s-t-a-t-e-`. Then, it removes the blank characters, resulting in `state`. The blank characters are essential to allow duplicate characters to be recognized correctly, by having a blank character between them. Optionally, this text prediction can be compared to a lexicon, which allows for additional validation and refinement. This comparison ensures that the recognized text aligns with known words or patterns, enhancing the accuracy and reliability of the OCR results. Throughout this thesis, we rely on version v1.6.2 of EasyOCR.

EasyOCR predicts a list of the text element bounding boxes, the confidence of the prediction and the text itself. The bounding boxes are defined by the (x, y) coordinates of their four corners. The text elements are a mix of words and multiple words, depending on the distance between the words. By default, the predictions are ordered from top to bottom and left to right. The approach of ordering text in lines of text can be effective when dealing with images that contain a single text column. However, when images have multiple text columns, enabling the paragraph mode can be more beneficial. In this mode, the model predicts the first column's text first and then proceeds to predict the second column, rather than switching between lines of both columns. Thus, by using paragraph mode, the model can better capture the structure and context of the text, leading to more accurate and coherent predictions for images with multiple text columns.

3.4 Tesseract

The second OCR method we utilize in this thesis is Tesseract OCR [1][41][42]. Originally developed by Hewlett-Packard in the 1980s, it was open sourced in 2005 and subsequently developed by Google from 2006 to 2018. Tesseract OCR employs adaptive thresholding to binarize the input image, enhancing the contrast between text and background. Following this step, the software performs page layout analysis to identify text regions within the image. These regions are then processed by the LSTM line recognizer to generate text predictions. Finally, the text prediction undergo correction measures to improve the accuracy of the result. To the best of our knowledge, the

literature on Tesseract OCR focuses on older versions of the software, predating the introduction of LSTM models with version 4.0.0. Consequently, detailed information about the LSTM line recognizer or any other replacement is not readily available. For our experiments in this thesis, we rely on version 4.1.1 of Tesseract OCR. Tesseract OCR offers multiple engine modes, namely the legacy engine only, the neural nets LSTM engine only or the Legacy + the LSTM engine. We use the pure LSTM engine mode, as it is the default and the most recent engine mode. Due to Tesseract OCR being implemented in C++, we utilize the Python wrapper, `pytesseract` [43], to conveniently incorporate it into our Python code.

To ensure a fair comparison, we do not preprocess the images for either OCR method, as we are unsure of the specific preprocessing steps employed by each method itself and wish to avoid any potential bias. For text prediction, we use the default settings for EasyOCR and Tesseract OCR. Comparative research between EasyOCR and Tesseract OCR, specifically in the task of recognizing text in license plate images [44], indicates that EasyOCR generally outperforms Tesseract OCR. Tesseract OCR, like EasyOCR, predicts a list of the text element bounding boxes, the confidence of the prediction and the text itself. It has a mode to predict paragraphs as well, however we found that it orders the predictions differently than EasyOCR. Thus, we use both methods to predict the raw bounding boxes with their text and sort them ourselves. This is necessary because the order of the predictions impacts the CER and we do not want to introduce any bias by using the order of one method over the other. In Figure 3.2 and Figure 3.3 we illustrate the predictions of both methods for the same image and attach the number in which the bounding boxes are ordered. EasyOCR already orders the predictions from left to right and top to bottom, as can be observed in Figure 3.2. Tesseract OCR, in contrast, interprets the left menu of the website as a paragraph, even without the paragraph mode, and predicts those text elements first, as illustrated in Figure 3.3. Note that the numbers for EasyOCR are lower, because it often predicts multiple words together. To bring the predictions in line with those of EasyOCR, we apply the following procedure to the predictions of Tesseract OCR. First, we find the top most bounding box in the image and save the y coordinates of the top half of that bounding box. Second, we identify all bounding boxes that overlap with those saved coordinates to gather all the text elements that are in the same line. Then, we sort those identified text elements from left to right, save them and remove them from the list of predictions. This procedure is repeated until all text elements are sorted. By following this procedure, the resulting order of predictions from Tesseract OCR becomes similar to EasyOCR and can be observed in Figure 3.4. The major difference

Figure 3.2: Unsorted EasyOCR predictions with order information

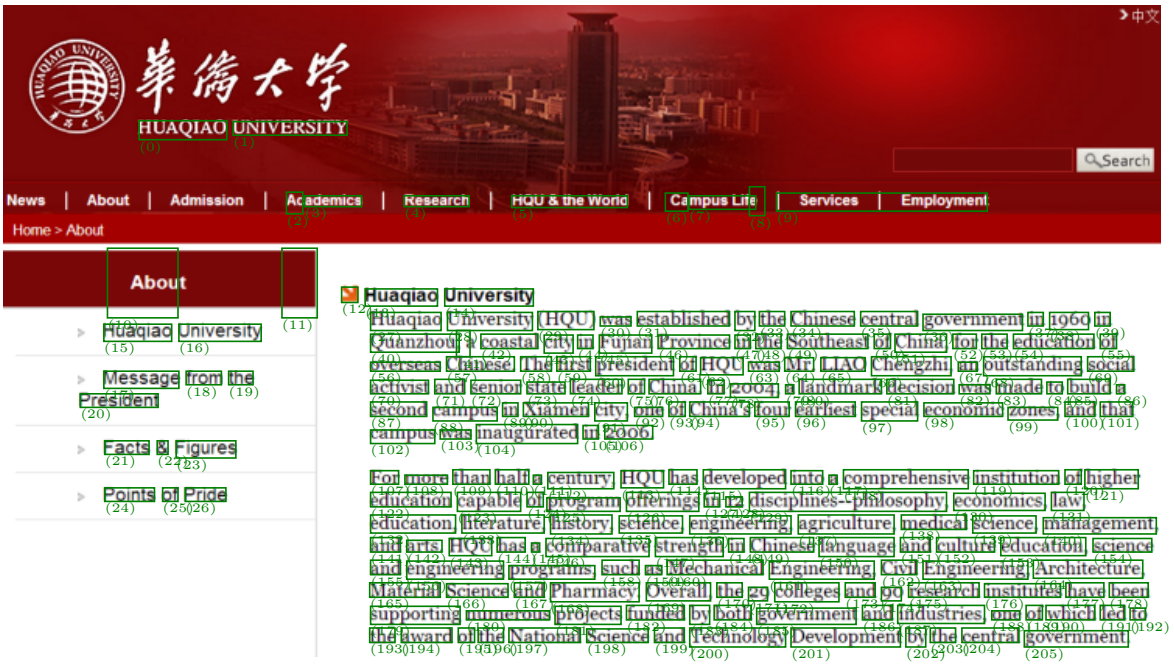


Figure 3.3: Unsorted Tesseract OCR predictions with order information

lies in the fact that Tesseract OCR tends to predict single words, whereas EasyOCR often predicts multiple words together. However, this distinction does not matter in our case since the predictions are combined into a single text prediction, by separating

Figure 3.4: Sorted Tesseract OCR predictions with order information

them with spaces, resulting in the same final text prediction. The intention behind this procedure is to ensure a unified order of the predictions for both methods. By doing so, we aim to prevent any disadvantage that could arise due to the order of their predictions, when comparing them to the true GT, which we introduce in section 4.2.

3.5 Character Error Rate

To evaluate the performance of OCR methods, we use the CER. The CER [45] describes how many substitutions, deletions, and insertions are necessary to transform a text prediction into the correct text label. It is defined as

$$\text{CER} = \frac{I + S + D}{N}, \tag{3.1}$$

with I being the number of insertions, S being the number of substitutions, D being the number of deletions, and N being the total number of characters in the text label. The CER ranges from 0 to ∞, where 0 means perfect recognition and the higher the worse the recognition. As an example, if the text label is `hello` and the prediction is `halo`, then the CER is $2/5 = 0.2$, due to one deletion (`l`) and one substitution (`a` $\rightarrow$ `e`).

In our analysis, we compare the CER to the subjective MOS. The details of the

subjective MOS will be discussed in the following chapter. Because the MOS is defined in the range 0 to 100 and 100 represents a high subjective quality, the two metrics are unintuitive to compare. Therefore, we take the complement of the CER by subtracting it from 1. Additionally we scale it by multiplying by 100 to get a MOS-like value, according to

$$\mathrm{CER_c} = (1 - \mathrm{CER}) \cdot 100. \tag{3.2}$$

This means that the $\mathrm{CER_c}$ can be in the range -∞ to 100, where 100 means perfect recognition and the lower the worse the recognition. However, in practice, the $\mathrm{CER_c}$ typically falls within the range of 0 to 100, which aligns with the scale of the MOS. This can be illustrated by a few examples. Let's consider the text label as `hello` and the prediction as empty. In this case, the CER would be $5/5 = 1$, as there are 5 insertions necessary to match the text label, and the $\mathrm{CER_c}$ would be 0. Thus, if the prediction is shorter than the text label, the $\mathrm{CER_c}$ cannot be negative. The only scenario where a negative $\mathrm{CER_c}$ can be obtained is if the prediction is longer than the text label and the prediction is incorrect. For instance, if the text label is `hello` and the prediction is `goodbye`, the CER would be $7/5 = 1.4$, as there are 5 substitutions and 2 deletions required to match the text label, resulting in a negative $\mathrm{CER_c}$ of -40. However, this case is highly unlikely, because the OCR methods incorporate a confidence threshold, which means that they refrain from predicting a word if they lack confidence in its correctness. Therefore, the word predictions are either empty, making the whole prediction shorter than the text label, or they have a high likelihood of being correct.

In order to apply the OCR methods discussed in this chapter, a suitable dataset comprising images containing text is essential. Thus, in the following chapter, we provide an overview of the dataset employed in this thesis.

Chapter 4

Dataset

In this chapter, we present the dataset used in our work. We use the screen content image database (SCID) dataset [3][1] as the base for our experiments. The SCID dataset is suitable for our work, since the images contain text on SCIs, different distortion levels and MOS values for each image. Among the available screen content datasets mentioned in the literature [6], we find that other options are either inaccessible or fail to fulfill all the necessary criteria we require for our research. An overview of the 40 reference images of the SCID dataset can be seen in Figure 4.1. Additionally, the dataset contains 1800 distorted images, which we discuss in the next section in detail. Further, MOS values are included for each of the distorted images, which represent the perceived quality by a human observer. The subjective tests to obtain the MOS values were conducted using the double stimulus method, involving the following steps. First, the reference image was shown to the candidates for 10 seconds, followed by a mid-gray screen. Afterwards, the distorted image was shown for 10 seconds. Finally, the candidates were asked to rate the distorted image's quality compared to the reference image on a 5-point scale, 1 being the worst and 5 being the best. These scores are then converted to a MOS value for each image between 0 and 100, with 100 representing the highest quality.

4.1 Distortion types

The 1800 distorted images are generated from the 40 reference images [3]. They are distorted with 9 different distortion types, each with 5 different distortion quality levels. In Table 4.1, we list the distortion types with a short description. The images distorted by Gaussian noise (GN) have noise added with zero mean and standard deviations of 0.001, 0.005, 0.01, 0.05 and 0.1 for each quality level, respectively. The images distorted by Gaussian blur (GB) are blurred with a Gaussian kernel. The size of the kernel is 5×5 with standard deviations of 0.58, 0.76, 0.96, 1.2 and 2.1 for

[1]The dataset can be downloaded here: https://eezkni.github.io/publications/ESIM.html.

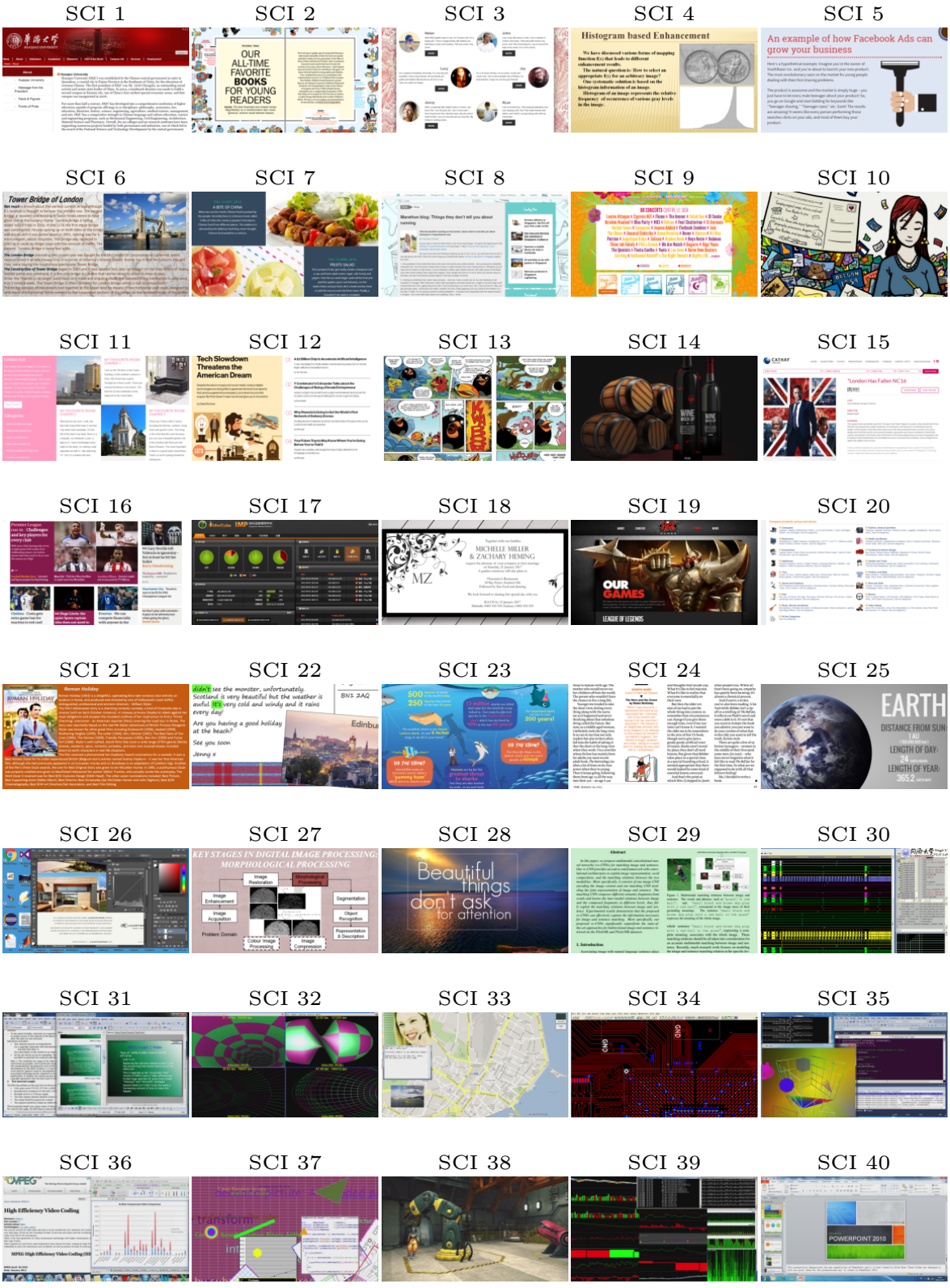


Figure 4.1: The 40 references images of the dataset.

Table 4.1: Overview of the distortion types used in the dataset.

Distortion Type	**Abbreviation**	**Description**
Gaussian Noise	GN	Addition of noise to an image using a Gaussian distribution
Gaussian Blur	GB	Blurring of an image using a Gaussian kernel
Motion Blur	MB	Blurring of an image due to movement of the camera or the object
Contrast Change	CC	Change in the contrast of an image
Joint Photographic Experts Group	JPEG	Image compression standard
Joint Photographic Experts Group 2000	JPEG2000	Image compression standard
Color Saturation Change	CSC	Changes in the color saturation of an image
High Efficiency Video Coding-Screen Content Coding	HEVC-SCC	Video compression standard for screen content
Color Quantization with Dithering	CQD	Reduction of colors available in an image

each quality level, respectively. Images distorted by motion blur (MB) are blurred with a motion kernel, which simulates motion blur. The parameter, which controls the degree of angle in a counter-clockwise direction, is set to zero and the parameter, which determines the length of the movement of the simulated camera, is set to 2, 3.4, 4, 5.5 and 6.4, respectively. The contrast change (CC) distortion scales certain pixel values in the reference image to new values to change the contrast. The scaling is applied for the ranges $[0, 1] \rightarrow [0.3, 0.5]$, $[0, 1] \rightarrow [0.1, 0.7]$, $[0.1, 0.8] \rightarrow [0.1, 0.9]$, $[0.2, 0.8] \rightarrow [0.1, 0.8]$ and $[0.2, 0.7] \rightarrow [0, 1]$, respectively. So for the first quality level, all pixel values (from 0 to 1) are scaled to values between 0.3 and 0.5 of the maximum pixel intensity. For the Joint Photographic Experts Group (JPEG) compression, the images are compressed by the image compression algorithm with quality factors 75, 35, 18, 8 and 5, respectively. The JPEG2000 compression is applied with compression ratios of 0.08, 0.045, 0.02, 0.015 and 0.01, respectively. The color saturation change (CSC) distortion keeps the luminance component of the images constant, but scales the chrominance components by the factors 0.96, 0.72, 0.58, 0.42 and 0.1, respectively. The HEVC-SCC distortion is applied by using the HEVC codec with the screen content coding (SCC) configuration on the images with the QPs set to 16, 36, 40, 42 and 48,

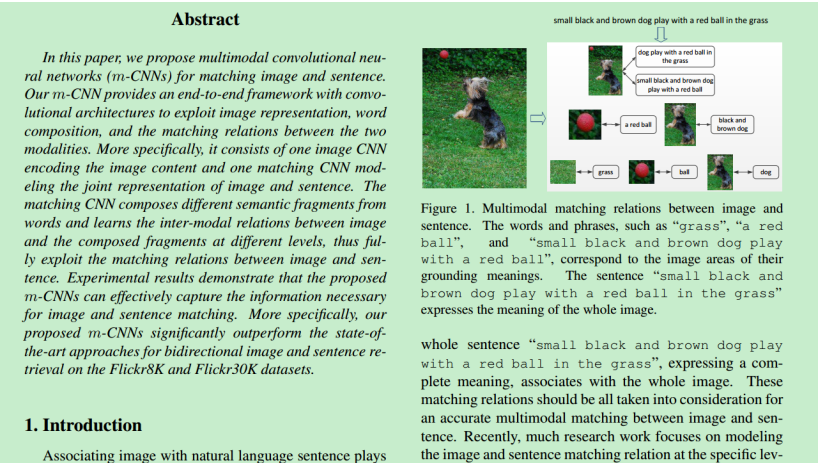


Figure 4.2: Reference image SCI29

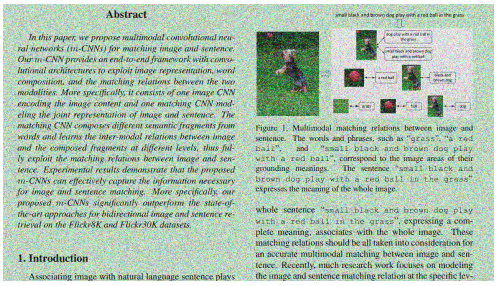
respectively. We are unsure which exact software version was used for the HEVC-SCC encoding. The color quantization dither (CQD) distortion is applied by reducing the number of colors available in the image to 30, 28, 25, 10 and 5, respectively. More detailed descriptions of the implementation of the distortions can be found in the supporting file included with the dataset. The different distortion types, at their most severe quality level, can be seen in Figure 4.3, applied to the image in Figure 4.2.

The impact of different distortion types on text within an image is evident. Among the various distortions, alterations in contrast or color have minimal effect on text legibility for human readers. Conversely, distortions such as GN, GB and MB can render the text completely unreadable to the human eye. We might expect that the distortions that affect the text for the human visual system the most, will also affect the OCR the most. It is to note, that all distortions are monotonically decreasing in their severity with the quality level[2] from 1 to 5. The only exception is CC. As illustrated in Figure 4.4, CC does not display a clear pattern of variation from low contrast to high contrast, or any similar trend. Understanding this behavior is important for the analysis of the trends of the MOS over the different quality levels.

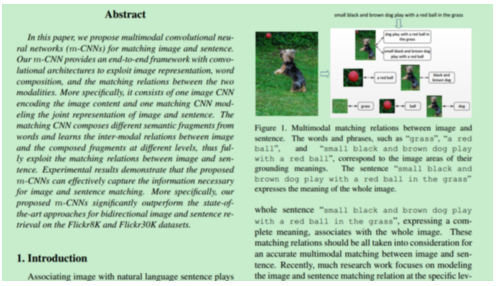
4.2 Labeling

Since we want to evaluate the OCR algorithms on theses images, we need a true GT in the form of a text label for each image, which are not contained in the dataset [3]. The following procedure is used to create the true GT for each image. We start by locating

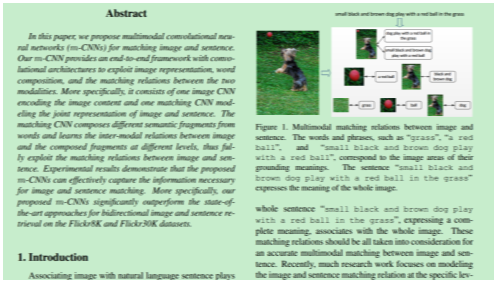
[2]It should be noted that referring to it as "quality level" might be somewhat counterintuitive, as the highest "quality level" (level 5) corresponds to the worst quality of the distorted image.



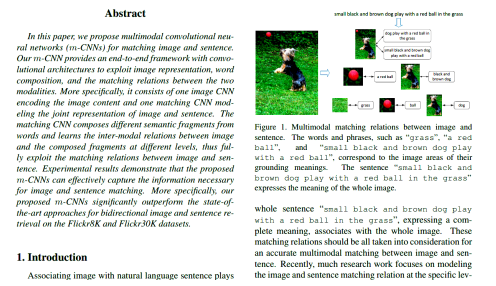
(a) Gaussian Noise



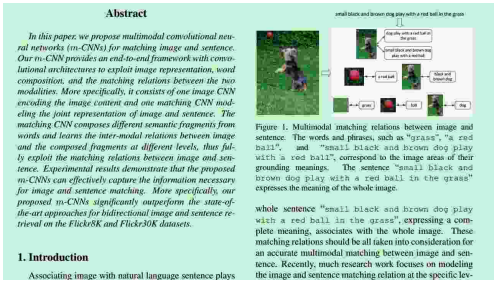
(b) Gaussian Blur



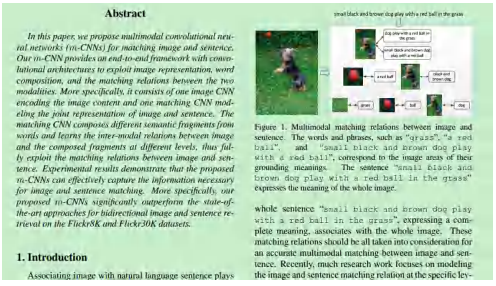
(c) Motion Blur



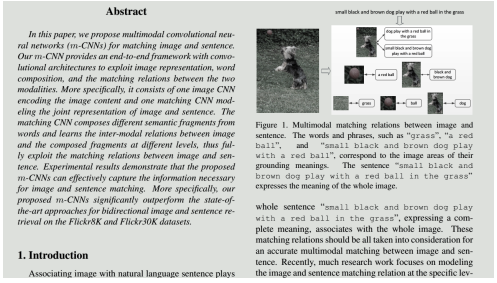
(d) Contrast Change



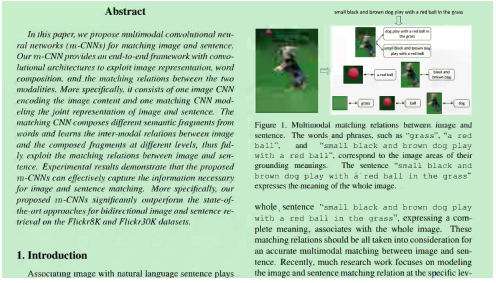
(e) JPEG Compression



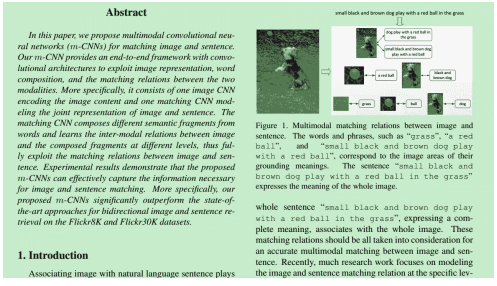
(f) JPEG2000 Compression



(g) Color Saturation Change



(h) HEVC Screen Content Coding



(i) Color Quantization with Dithering

Figure 4.3: SCI 29 distorted by 9 different distortion types at the most severe level.

Figure 4.4: Distorted image 1 with different levels of CC, quality levels from left to right: 1, 2, 3, 4, 5.

the topmost word in the image. Then, we identify if this word is part of a line. If it is, we record the text of the entire line. Afterwards, we move to the next line or word and repeat the process until all text elements are recorded. Finally, we combine all the recorded text elements into the full true GT by separating them with spaces. In this process, we ignore paragraphs and only consider the vertical position of the lines. This true GT aligns with the prediction order of the OCR algorithms, as discussed in section 3.4. To avoid introducing bias towards any specific OCR algorithm regarding the order of text elements, we made the decision not to utilize one of the algorithms for predicting the text and then correcting it to create the true GT, but fully label them ourself.

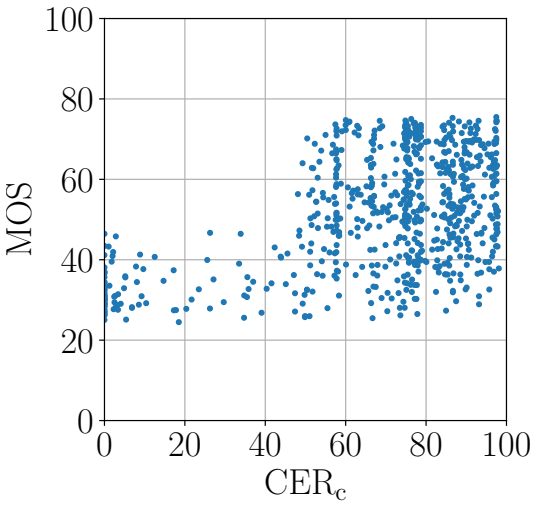
4.3 Analysis

Before we continue with our main experiments, we give a short analysis of the dataset. First, we select a subset of images for our experiments. Some of the images contain no text at all or only numbers. Others contain some text, but have a large focus on graphical objects besides the text. This makes them less suitable for a comparison between the CER_c, which evaluates text, with the MOS, which evaluates the whole image. Due to these factors, for our experiments that involve the MOS, we select the images with

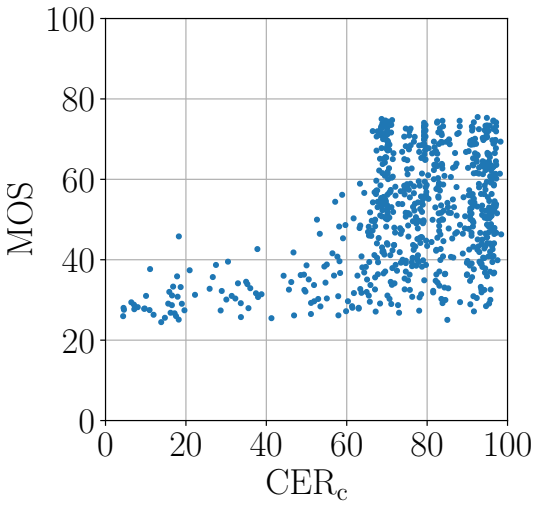
$$i \in \{1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 15, 18, 20, 21, 24, 29\}, \tag{4.1}$$

as their main focus lies on text elements, instead of graphical objects. This selection process is subjective, so it might be more reasonable to use all images and filter out outliers later. Additionally, even if an image only has one small text element, the CER_c might still be a good estimation of the MOS, if the distortion affects the text in the same way as the rest of the image. This however, is not the case for certain distortions, like JPEG or other compression algorithms.

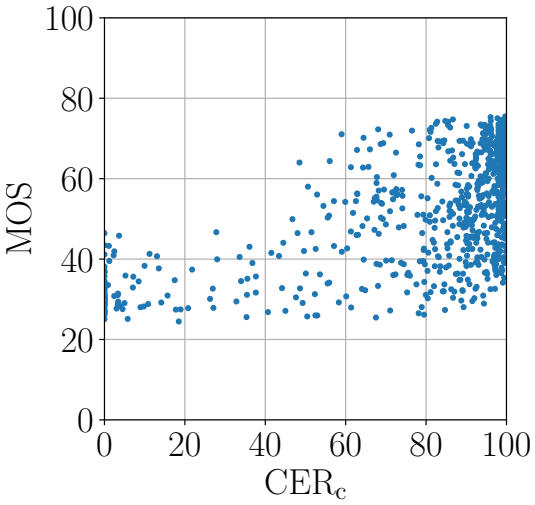
In Figure 4.5 the CER_c is plotted against the MOS for both OCR algorithms and both GTs. Generally, we observe that the MOS ranges from 20 to 80 for all figures.



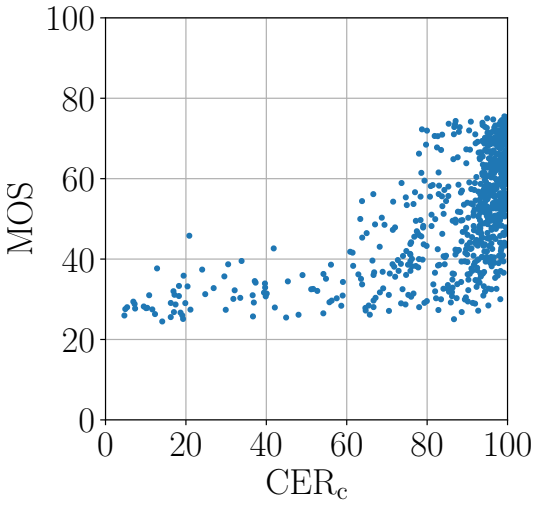
(a) CER vs MOS for Tesseract OCR and the true GT.



(b) CER vs MOS for EasyOCR and the true GT.



(c) CER vs MOS for Tesseract OCR and the pseudo GT.



(d) CER vs MOS for EasyOCR and the pseudo GT.

Figure 4.5: CER_c vs MOS for Tesseract OCR and EasyOCR, with the true GT and the pseudo GT.

On the other hand, the CER_c spans from 0 to 100 for all figures. Comparing Tesseract OCR with EasyOCR for the true GT, we observe that the CER_c distribution for Tesseract OCR is more spread out with more lower CER_c values compared to EasyOCR Additionally, there are some points with zero CER_c for Tesseract OCR. This implies that EasyOCR performs better in general and that Tesseract OCR struggles with some distortions and fails to predict anything. We notice similar behavior when comparing the pseudo GTs for both OCR algorithms, although the CER_c values are generally higher. This can be attributed to the fact that the predictions contain no additional errors due to positioning of the text elements and are generally closer to the pseudo GT compared to the true GT. Lastly, it is worth noting that there are minimal occurrences of high MOS values paired with low CER_c values, which is evident in the top left sections of all figures. In the following section we detail the extension of the dataset with compressed images that are used to compare two video codecs.

4.4 Extension of the Dataset with Images Distorted by Compression Methods

In this section, we first introduce the two video codecs, the HEVC and the VVC. We further, explain how the two codecs are adjusted for SCIs by using screen content extensions. Finally, we use these codecs for the extension of the dataset by encoding the reference images.

4.4.1 High Efficiency Video Coding

The HEVC [4] is one of the newest video codecs. It is the successor of the H.264/MPEG-4 AVC codec. The main improvements were the leveraging of parallel processing architecture in modern devices and addressing higher resolutions. The codec uses the conventional approach of dividing the image into block shaped regions. The information about the block size is added to the bit stream sent to the decoder. The first image of a video sequence uses intraframe prediction, which uses information from neighboring blocks to predict the information in the current block. For further frames, interframe prediction is used, which leverages the difference from the previous frame to encode the current frame. This improves coding efficiency, since the difference between frames is usually cheaper to encode than a whole new frame. However, in our case we only encode single images, so the codecs only use the intraframe prediction. After predicting the current block, the residual, which represents the difference between

the prediction and the original block, is transformed by a linear spatial transform to generate the transform coefficients. Those are scaled, quantized and entropy coded to further reduce the bit rate. The prediction information, the transform coefficients and all other required information is then sent to the decoder. The decoder uses that information to predict the current block as well by replicating the encoder. Afterwards, the transform coefficients can be reconstructed and used to approximate the residual of the block. The residual is then added to the prediction to reconstruct the original block. Further, it employs a number of new techniques over its predecessors to provide around 50% bit rate savings for equivalent quality. To conform to the special characteristics of SCIs, a screen content extension was developed for the HEVC [46]. We will expand on this in the following subsection after we introduce the VVC, as they share some similarities. For this thesis, we use version 16.21+SCM-8.8 of the HM reference software [47] to encode the images with the HEVC codec. The default [48] and SCC [49] configurations used for encoding are applied according to the common test conditions for color space RGB 444.

4.4.2 Versatile Video Coding

The VVC [5] is the successor of the HEVC and one of the most recent video codecs. Compared to the HEVC, it introduces combined inter-/intraframe prediction, luma mapping with chroma scaling and additional loop filters. Additionally, while most implementations of the HEVC are only able to use square block sizes, the VVC supports rectangular block sizes as well, which enables more efficient coverage of regions that can be encoded efficiently. It aims to reach another 50% bit rate savings compared to the HEVC for equivalent quality. Further, the versatility of the codec enables it to be used for a wider range of applications, including 360° immersive video, high dynamic range, adaptive streaming with resolution changes and many more. For this thesis, we use version 17.2 of the VTM reference software [50] to encode the images with the VVC codec. The default and SCC [51] configurations used for encoding are applied according to the common test conditions for color space RGB 444.

For SCIs there are some additional tools [5] to improve the performance of the codecs due to the different characteristics of the images. One such tool is the palette mode, which uses a reduced number of colors to encode blocks of images, because SCIs generally contain a limited amount of colors in local regions. This tool exists for the HEVC, but is further improved in the VVC. Another tool is the intra-picture block copy, which enables the codecs to use a copy of a block as the prediction for

(a) Default configuration (b) SCC configuration

Figure 4.6: Normalized absolute pixel differences between the reference image and the HEVC encoded images with the default and SCC configurations for Image 4.

(a) Default configuration (b) SCC configuration

Figure 4.7: Normalized absolute pixel differences between the reference image and the VVC encoded images with the default and SCC configurations for Image 4.

another block, It leverages the fact that SCIs often contain repeated patterns, for instance in the form of UI elements or large uniformly colored regions. In the HEVC screen content extension, this tool is able to copy blocks from the same frame from further away, while in the VVC the complexity is reduced by restricting the copying to neighboring blocks. The improvements from the screen content tools for HEVC are evident when observing Figure 4.6. The figures depict the normalized absolute pixel differences between a reference image and its corresponding encoded image, with brighter pixels representing a larger difference. Notably, the absolute pixel differences, representing the coding error, are reduced for the SCC configuration compared to the default configuration, particularly in the text regions. A similar, albeit more subtle difference is observable for VVC in Figure 4.7.

It is to note, that we use these codecs on images instead of videos, which implies that we are not leveraging the full potential of the videos codecs. To extend the dataset

we encode the reference images with the default and the SCC configuration of the codecs. The most important difference to the other distorted images is that there are no subjective scores available for these images. However, we can use more images for the comparison of the codecs, as we do not need to select based on too much focus on graphical objects over the text elements. For the experiments related to the codecs we select the images with

$$i \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 15, 16, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29\}. \quad (4.2)$$

The common test condition QPs $\in \{22, 27, 32, 37\}$ result in no significant changes in the $\mathrm{CER_c}$. Therefore, following the approach of previous researchers in [52], we encode these images with QPs $\in \{35, 40, 45, 50\}$ for both codecs. We subsequently employ the OCR algorithms to extract text from these images, enabling us to compute the $\mathrm{CER_c}$. Afterwards, we can visualize rate-distortion curves and calculate the BDRate, as described in section 2.4.

In summary, the original dataset contains a MOS for each image with various distortions. To evaluate the performance of the OCR algorithms, we create text labels for each of the images. Further, we expand the dataset by encoding select reference images with the HEVC and VVC codecs. With all the necessary components to describe the experiments now in place, we proceed to evaluate and discuss our results in the subsequent chapter.

Chapter 5

Evaluation

In this chapter, we evaluate the results of our experiments in the following three sections. First, we analyze the performance of the OCR algorithms for each distortion type against the true GT. Then, we compare the performance of the OCR algorithms against human judgment against the pseudo GT. Finally, we investigate the feasibility of using the recognized text as a pseudo GT to compare the performance of video codecs.

5.1 Performance of Optical Character Recognition

First, we assess the impact of different distortion types on OCR performance by comparing the $\overline{\text{CER}}_\text{c}$ for various quality levels. In this section, the CER_c is calculated in relation to the true GT for each image. Additionally, we use the selection of images defined in section 4.3 for this section. We perform this analysis separately for EasyOCR and Tesseract OCR to compare their performance. Additionally, we conduct this analysis separately for all nine distortion types. For the comparison, we plot the $\overline{\text{CER}}_c$ in relation to the true GT on the y-axis, against different quality levels of the distortions on the x-axis.

In Figure 5.1, we depict the $\overline{\text{CER}}_\text{c}$ in relation to the true GT for different quality levels using EasyOCR. We notice a trend that MB has the most significant impact on EasyOCR's performance, exhibiting a nearly linear decrease from 80 to 20. For JPEG and GB, EasyOCR displays similar behavior until quality level 4, after which it experiences a steeper decline for GB. This may be due to the blurring of images greatly affecting the legibility of text, with letters becoming indistinct and merging together. For GN, JPEG2000 and HEVC-SCC, EasyOCR exhibits similar performance, experiencing a slight decline at quality level 5. The remaining distortions have minimal impact on performance, which we expect since color distortions do not directly affect the text shapes. However, one might expect that the change in contrast impacts OCR performance since it is more difficult to differentiate between text and background.

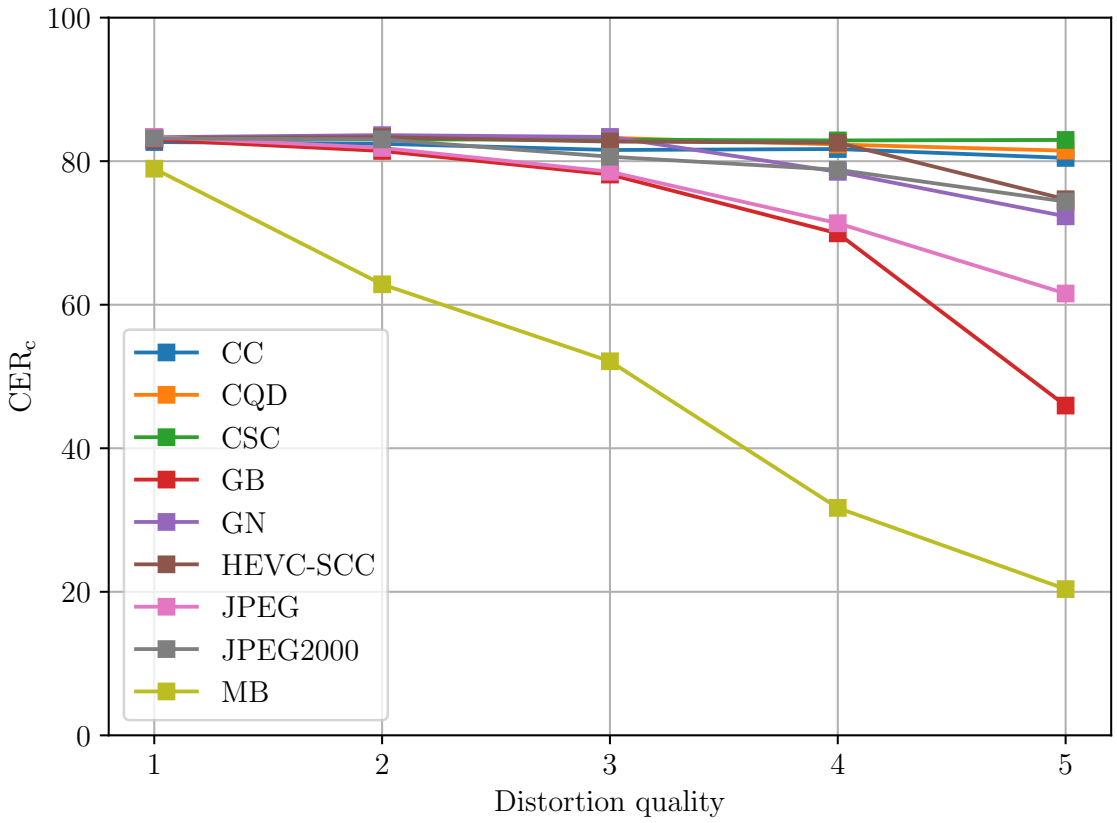


Figure 5.1: $\overline{\mathrm{CER}_c}$ in relation to the true GT for different quality levels with EasyOCR.

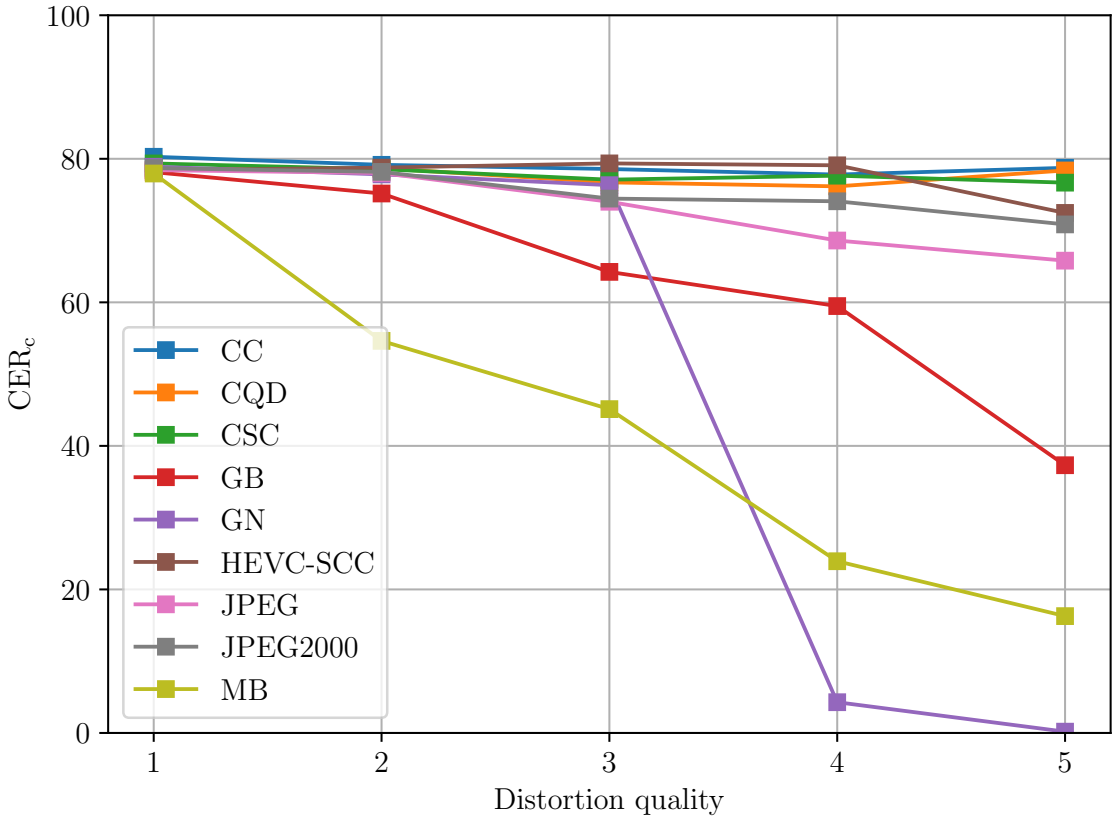


Figure 5.2: $\overline{\text{CER}}_c$ in relation to the true GT for different quality levels with Tesseract OCR.

One possible explanation for this phenomenon could be that the contrast remains sufficiently high, thereby not affecting EasyOCR's performance.

In Figure 5.2, we can see the same analysis for Tesseract OCR. We see a steady decrease in $\overline{\text{CER}}_c$ for MB and GB, with the former exhibiting a steeper decline. However, the most noteworthy finding is the significant drop in $\overline{\text{CER}}_c$ for GN at quality level 4, reaching 0 at quality level 5. This sudden decline is particularly surprising when compared to other types of distortion. Among the distortions analyzed, CC, CQD, and CSC do not impact the performance. The results for CC even demonstrate a slightly higher $\overline{\text{CER}}_c$ at quality level 5 compared to previous levels. This phenomenon can possibly be attributed to CC's ability to enhance the distinction between the text and the background, acting almost like an image preprocessing step for the OCR. Additionally, as we saw in Figure 4.4, CC does not follow a clear trend with decreasing quality level. Thus, it is difficult to interpret a trend for CC.

In summary, our results reveal that MB and GB have a substantial impact on the

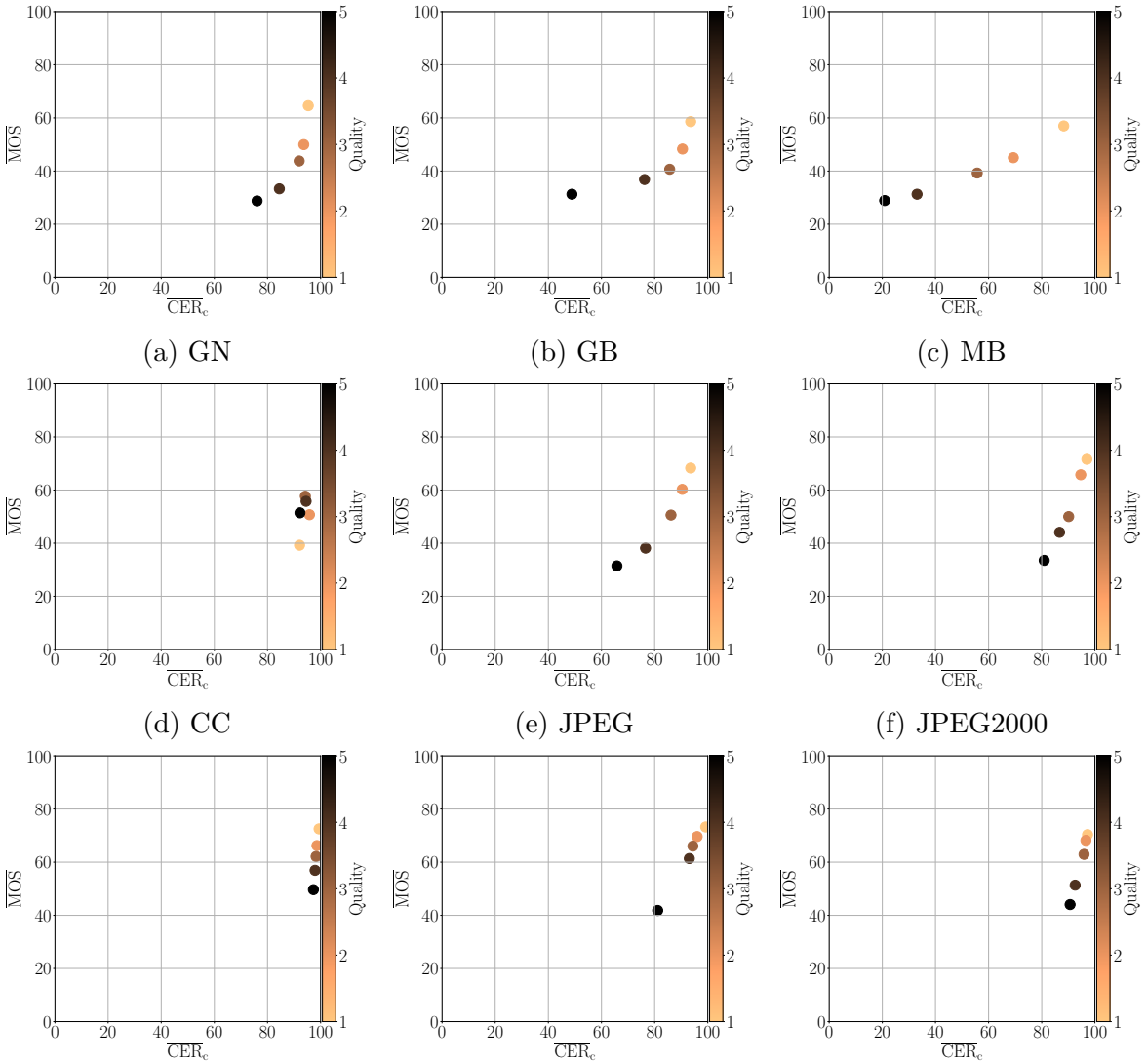
performance of both EasyOCR and Tesseract OCR algorithms. Additionally, both OCR algorithms demonstrate superior performance across images with CC, CQD, and CSC. However, the most remarkable discovery is the significant drop in the $\overline{\text{CER}}_\text{c}$ for Gaussian noise at quality level 4, even reaching 0 at quality level 5, for Tesseract OCR. This suggests that EasyOCR exhibits greater robustness to GN compared to Tesseract OCR. Overall, EasyOCR outperforms Tesseract OCR, with the highest $\overline{\text{CER}}_\text{c}$ of approximately 83 for EasyOCR, while Tesseract OCR only achieves around 80.

5.2 Comparison With Human Judgment

In this section, we compare the performance of the OCR algorithms with human judgment. To visualize this, we plot the $\overline{\text{CER}}_\text{c}$ on the x-axis against the $\overline{\text{MOS}}$ on the y-axis for each distortion type. We create seperate plots for EasyOCR and Tesseract OCR. For our analysis, we use the image selection defined in section 4.3. We calculate the mean for each metric over all images for each quality level and distortion type separately. Moreover, unlike the last section, we compute the CER_c with respect to the pseudo GT. This provides a fairer comparison since the MOS is determined by humans comparing the distorted images to the reference images.

In Figure 5.3, we can observe the $\overline{\text{CER}}_\text{c}$ in relation the pseudo GT against the $\overline{\text{MOS}}$ for all distortions and qualities for EasyOCR. In general we notice that the performance ceiling for the $\overline{\text{CER}}_\text{c}$ is now up to almost 100. This is due to the $\overline{\text{CER}}_\text{c}$ being calculated in relation to the pseudo GT, which is not necessarily the true GT. The distortions CSC and CC are not impacting the performance of EasyOCR much, like we saw in the previous section. However, the corresponding $\overline{\text{MOS}}$ values are impacted. For CC, we can see that the $\overline{\text{MOS}}$ does not show a clear trend with the quality level, because the CC does not either, like we mention in section 4.1. For all other distortions the graph shows a clear trend, where both $\overline{\text{CER}}_\text{c}$ and $\overline{\text{MOS}}$ decline with decreasing quality. This implies at least a slight correlation between the two metrics, which we will quantify later.

In Figure 5.4, we conduct the same analysis for Tesseract. Tesseract OCR performs generally worse on all distortions compared to EasyOCR. Due to the $\overline{\text{MOS}}$ generally showing a steeper decline compared to the $\overline{\text{CER}}_\text{c}$, for some distortions the CER_c from Tesseract OCR might exhibit a higher correlation with the MOS. For instance, Tesseract's performance on images impacted by JPEG seems almost perfectly linear, compared to EasyOCR's nonlinear performance. We can clearly see that the $\overline{\text{CER}}_\text{c}$ drops sharply on quality levels 4 and 5 of GN while the $\overline{\text{MOS}}$ does not. Such a large

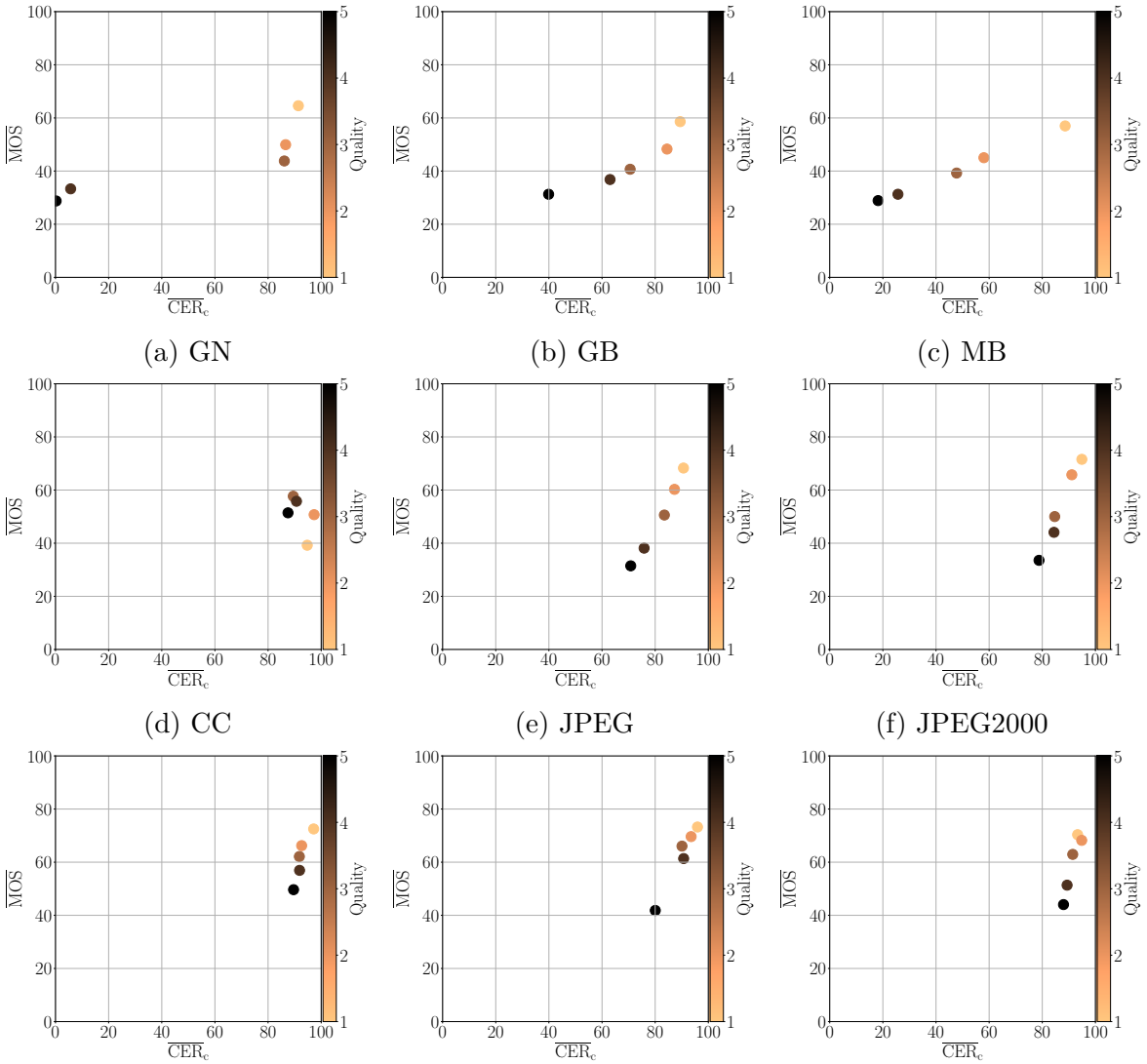


(a) GN (b) GB (c) MB

(d) CC (e) JPEG (f) JPEG2000

(g) CSC (h) HEVC-SCC (i) CQD

Figure 5.3: $\overline{\text{CER}}_c$ in relation to the pseudo GT against the $\overline{\text{MOS}}$ for different distortion types and quality levels with EasyOCR.



(a) GN (b) GB (c) MB

(d) CC (e) JPEG (f) JPEG2000

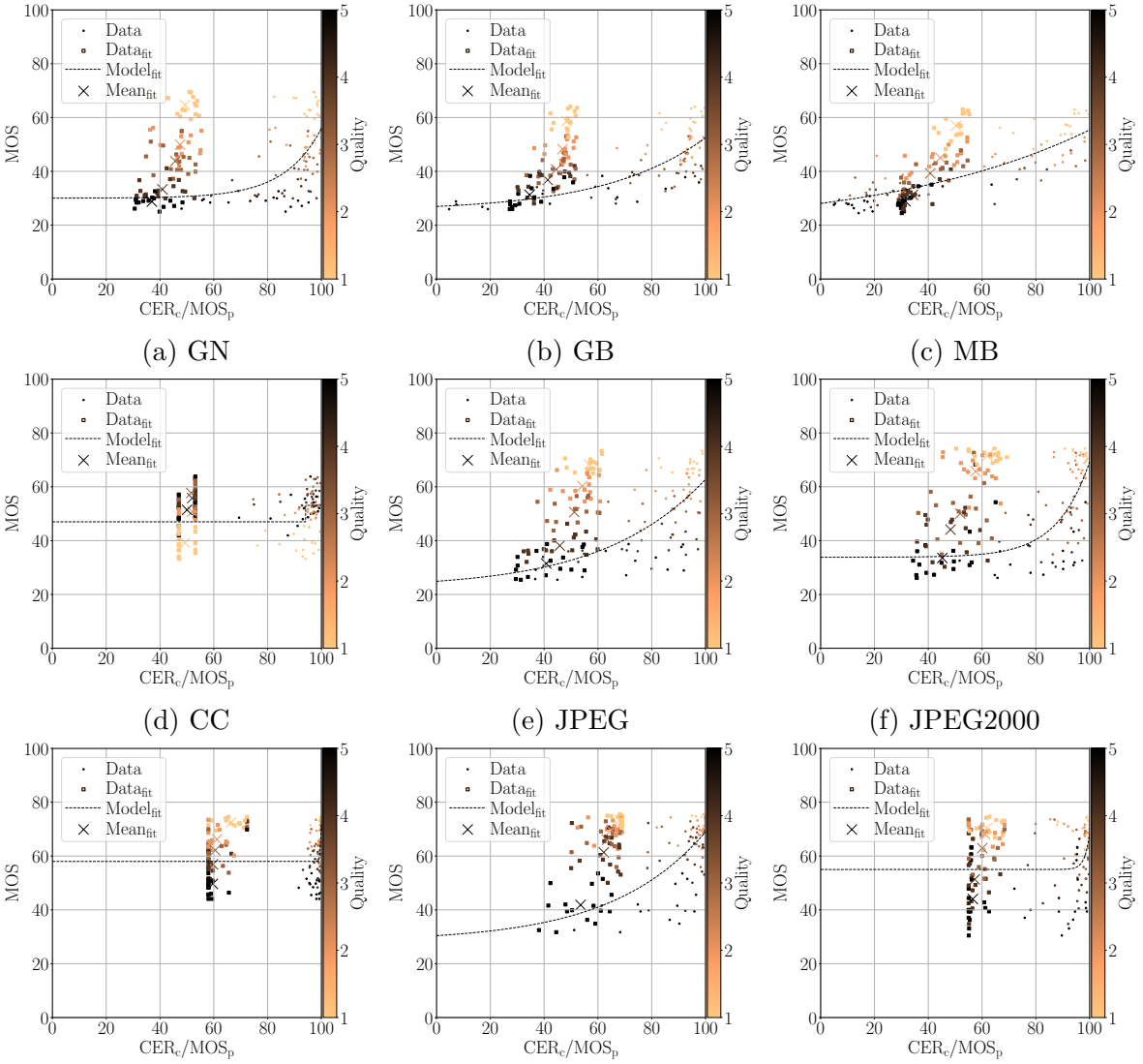
(g) CSC (h) HEVC-SCC (i) CQD

Figure 5.4: $\overline{\text{CER}_c}$ in relation to the pseudo GT against the $\overline{\text{MOS}}$ for different distortion types and quality levels with Tesseract OCR.

drop is not representative of the trend of the MOS and is therefore not desirable. In general however, we need to be careful, as we are only considering the mean values of the CER_c and MOS. For a full analysis the full distribution of the metrics needs to be considered. In the next part, we will consider all datapoints separately. Additionally, we fit a model to the data points to remove nonlinearities as outlined in subsection 2.3.1.

In Figure 5.5, we can see the single CER_c values plotted against the MOS for all distortions with EasyOCR. The graph also includes the fitted model. Lastly, the resulting predicted MOS_p values and their mean over each quality are depicted in relation to the MOS. The x-axis shows either the CER_c or the modified values as MOS_p. First, we can observe that the models fit well to the data points and are monotonic. The resulting MOS_p values are generally moved to the center compared to the original CER_c values. This makes them have a more linear relationship to the MOS values. For CC, CSC and CQD, we can see that, when the original points are very close to each other, it is difficult to fit a good model. Thus, the resulting MOS_p values seem to be almost constant, which is undesirable as they lose their predictive value, because the MOS values are obviously not constant. As we mentioned before, this is expected due to the superior performance of EasyOCR on these distortions for all quality levels, while the MOS varies. We depict the same analysis for Tesseract OCR in Figure 5.6. Compared to the fitting for EasyOCR, it is difficult to see any meaningful differences. One noticeable difference are the now also relatively constant MOS_p values for GN and GB. However to quantify and compare our results, we look at the correlations and the RMSE in the next part.

To quantify the observations we calculate the SRCC (prediction monotonicity), PLCC (prediction consistency) and RMSE (prediction accuracy) as described in chapter 2. In Table 5.1, we can see metrics for each distortion separately and for all distortions together (overall) for EasyOCR. Most noteworthy, MB shows the highest SRCC and PLCC and the lowest RMSE. This means that the predicted MOS_p values are a decent estimation of the MOS values. Second, the GB distortion shows the second highest SRCC and PLCC and the second lowest RMSE. However, it is 15 points below the MB distortion in SRCC and PLCC and 1.67 points higher in RMSE. Thus, it is a significantly worse estimation of the MOS values compared to images with MB. On the other hand, we can see that the performance for CC, CQD and CSC show the lowest SRCC and PLCC. So for those distortions the predicted MOS_p values are a bad estimation of the MOS values. The overall performance with a SRCC of 0.65, a PLCC of 0.66 and a RMSE of 9.44 is not high enough to support a general recommendation

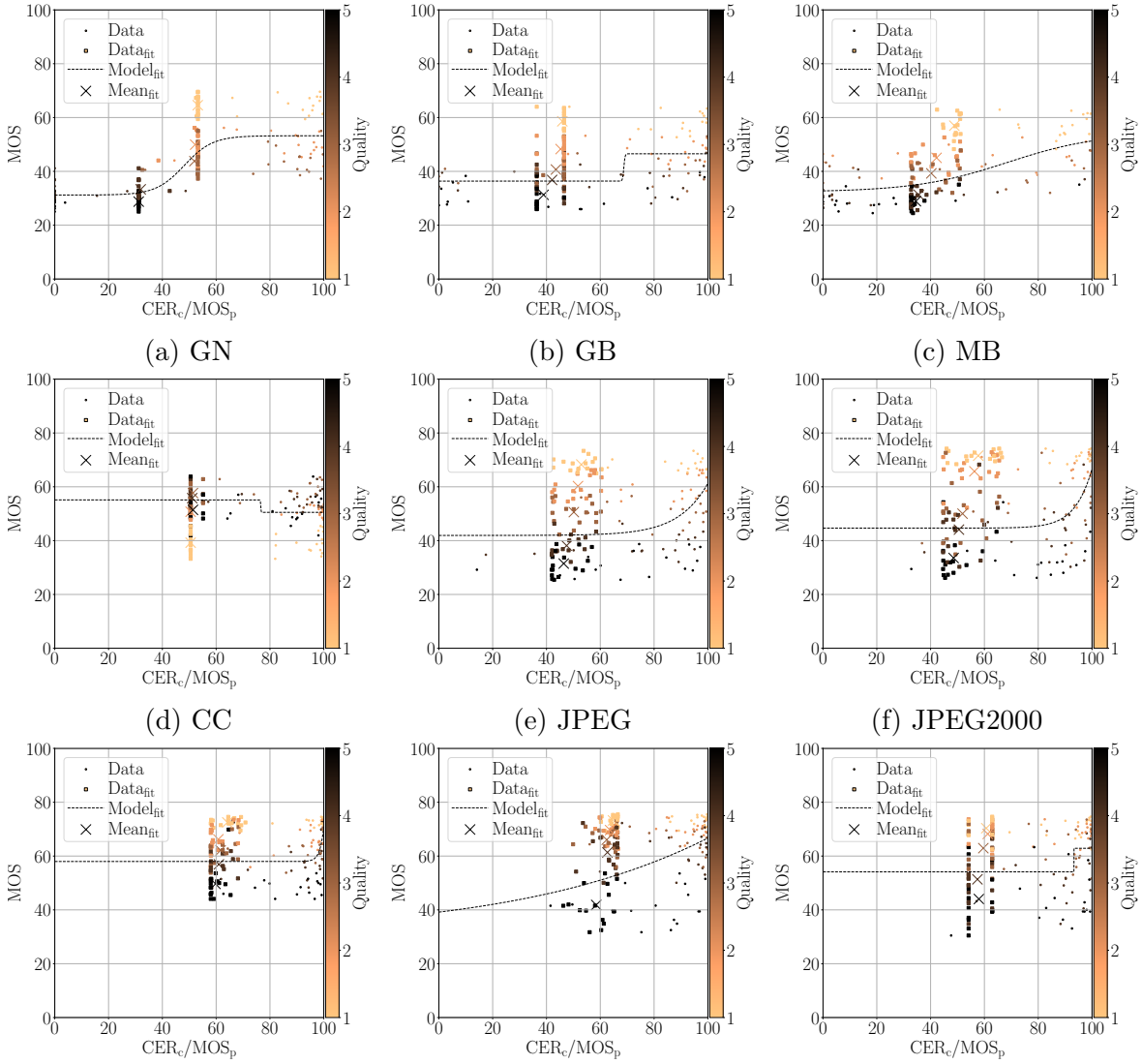


(a) GN (b) GB (c) MB

(d) CC (e) JPEG (f) JPEG2000

(g) CSC (h) HEVC-SCC (i) CQD

Figure 5.5: Single datapoints (CER_c vs MOS), the fitted model, the single datapoints after fitting (MOS_p vs MOS) and the mean over each quality after fitting (MOS_p vs MOS); all in relation to the text predictions on the reference images for different distortion types with EasyOCR.



(a) GN (b) GB (c) MB

(d) CC (e) JPEG (f) JPEG2000

(g) CSC (h) HEVC-SCC (i) CQD

Figure 5.6: Single datapoints (CER_c vs MOS), the fitted model, the single datapoints after fitting (MOS_p vs MOS) and the mean over each quality after fitting (MOS_p vs MOS); all in relation to the text predictions on the reference images for different distortion types with Tesseract OCR.

Distortion Type	SRCC	PLCC	RMSE
CC	0.29	0.40	6.76
CQD	0.28	0.36	11.45
CSC	0.47	0.54	7.65
GB	**0.70**	**0.71**	7.40
GN	0.56	0.54	11.09
HEVC-SCC	0.57	0.62	9.42
JPEG	0.65	0.66	10.65
JPEG2000	0.59	0.61	12.41
MB	**0.86**	**0.85**	**5.73**
Overall	0.65	0.66	9.44

Table 5.1: SRCC between CER_c and MOS; PLCC and RMSE between MOS_p and MOS for different distortion types and for all distortions together (overall) for EasyOCR.

for using EasyOCR as a estimation of the MOS. Nevertheless, these findings suggest that EasyOCR is a good approximation for human quality perception if images are impacted by blur. Additionally, it might be an attractive addition to other combined quality metrics, where only text regions are assessed by OCR and pictorial regions are estimated by other IQA methods.

In Table 5.2, we can see the same metrics for Tesseract OCR. The performance on images with MB shows a lower SRCC and PLCC and a higher RMSE compared to EasyOCR. The three metrics are even worse than the performance on images with GB for EasyOCR. In general, we notice that the ability of Tesseract OCR to approximate human quality perception is worse than EasyOCR for almost all distortions. The only exception are images with GN, where Tesseract OCR shows a higher SRCC and PLCC and a lower RMSE than EasyOCR. However, this is not meaningful as the MOS_p values for GN are mostly constant, see Figure 5.6a, and thus lose their predictive value. The overall performance with SRCC of 0.55, PLCC of 0.59 and RMSE of 10.17 is not high enough to support a general recommendation for using Tesseract OCR as a approximation of the MOS either.

To summarize, in general EasyOCR is a better approximation of human quality perception compared to Tesseract OCR. Additionally, with a SRCC of 0.86 and a PLCC of 0.87, EasyOCR shows potential as a rough estimate of the MOS for textual images with MB. However, specifically in online communication, images are not very often impacted by blurring. One application might be the IQA of images of street signs with text from self driving cars or scrolled text in video conferences, as they deal

Distortion Type	SRCC	PLCC	RMSE
CC	-0.15	0.19	**7.24**
CQD	0.31	0.35	11.51
CSC	0.47	0.44	8.17
GB	0.53	0.46	9.30
GN	**0.77**	**0.81**	**7.77**
HEVC-SCC	0.30	0.40	11.02
JPEG	0.45	0.44	12.66
JPEG2000	0.48	0.49	13.70
MB	0.64	0.69	**7.90**
Overall	0.55	0.59	10.17

Table 5.2: SRCC between CER_c and MOS; PLCC and RMSE between MOS_p and MOS for different distortion types and for all distortions together (overall) for Tesseract OCR.

with MB. Finally, we can compare our results to other IQA algorithms evaluated on the same dataset summarized in [3]. Most of the algorithms show a overall SRCC and PLCC above 0.75 with some reaching 0.85. Our overall performance for EasyOCR and Tesseract OCR is significantly lower around 0.55 to 0.65. Our overall RMSE is on the higher end compared to the other algorithms. However the performance for MB is close to some of the best performing algorithms [53] with a SRCC of around 0.9, a PLCC of around 0.91 and a RMSE of around 4.4. It is important to note that we select a subset of the dataset for our experiments, which has a positive impact on the performance of our method, as some images do not contain text at all and would result in no quality assessment. On the other hand, the other methods were able to take the entire images into account, while our methods only use the textual regions, which miss valuable information for distortions that do not affect these regions much, like CQD. Thus, the comparison is not entirely accurate.

5.3 Usage of Recognized Text as Ground Truth

In this section, we evaluate the feasibility of using recognized text by the OCR algorithms as the true GT. To do so, we use both EasyOCR and Tesseract OCR to recognize the text in the reference images without distortions. Then, we compare the recognized text, the pseudo GT, with the hand annotated true GT to calculate the CER_c. For this analysis, we still use the selection of images defined in section 4.3.

From Table 5.3, we can see that the $\overline{CER_c}$ for EasyOCR is roughly 4 points higher compared to Tesseract OCR. With a roughly 2 points higher standard deviation, the

	CER_c	
OCR Algorithm	Mean	Std. Dev.
EasyOCR	83.25	10.26
Tesseract	79.08	12.00

Table 5.3: Mean and standard deviation of CER_c for the predictions of EasyOCR and Tesseract OCR over selected reference images compared to the true GT.

performance of Tesseract OCR is also more inconsistent. With a $\overline{CER}_c$ of 83.79 its difficult to recommend using EasyOCR as a true GT source, as almost 16% of the text might be wrong. Tesseract is worse with a $\overline{CER}_c$ of 79.1, and thus cannot be recommended either. It might however be possible to improve the performance of the OCR algorithms by applying a preprocessing pipeline on the images before deploying the OCR methods.

Although the performance of the OCR algorithms is not good enough to be used as a true GT source, we can still use it to compare the performance of different codecs. Thus, we now focus on the performance of the OCR algorithms on images encoded with the HEVC and VVC. For this experiment we refer to the HEVC as HM and the VVC as VTM, their respective reference software. Additionally, we modify the dataset for the codec comparison and use the more expansive selection of images, as detailed in section 4.4. To be clear, in this section, we refer to the hand annotated GT as the true GT. The recognized text by the OCR algorithms on the reference images is referred to as pseudo GT. The goal is to determine if the OCR algorithms can be used as a pseudo GT for the comparison of different codecs. More specifically, if the difference between both pseudo GTs rate-distortion curves is similar to the difference between both true GTs rate-distortion curves, we might be able to use the pseudo GT to compare the performance of other, similar codecs. The following plots have the same structure. They show multiple rate-distortion curves for different codecs. On the x-axis we see the bitrate of the compressed images in Mbit/s. The y-axis shows the $\overline{CER}_c$. We plot the mean of the different QPs for each codec for the pseudo and the true GT.

In Figure 5.7, we can see the $\overline{CER}_c$ in relation to the pseudo and the true GT against the bitrate of the images for the HM and VTM codec with the default codec configuration for EasyOCR. We can observe that the VTM curves are generally further to the top left corner, compared to the HM curves. This indicates that EasyOCR performs better on the VTM encoded images and they require less bitrate. Additionally, we can see that the trends of the true GT curve pair (blue) are similar to the trends of the pseudo GT curve pair (red). In Figure 5.8, we can see the same curves with the

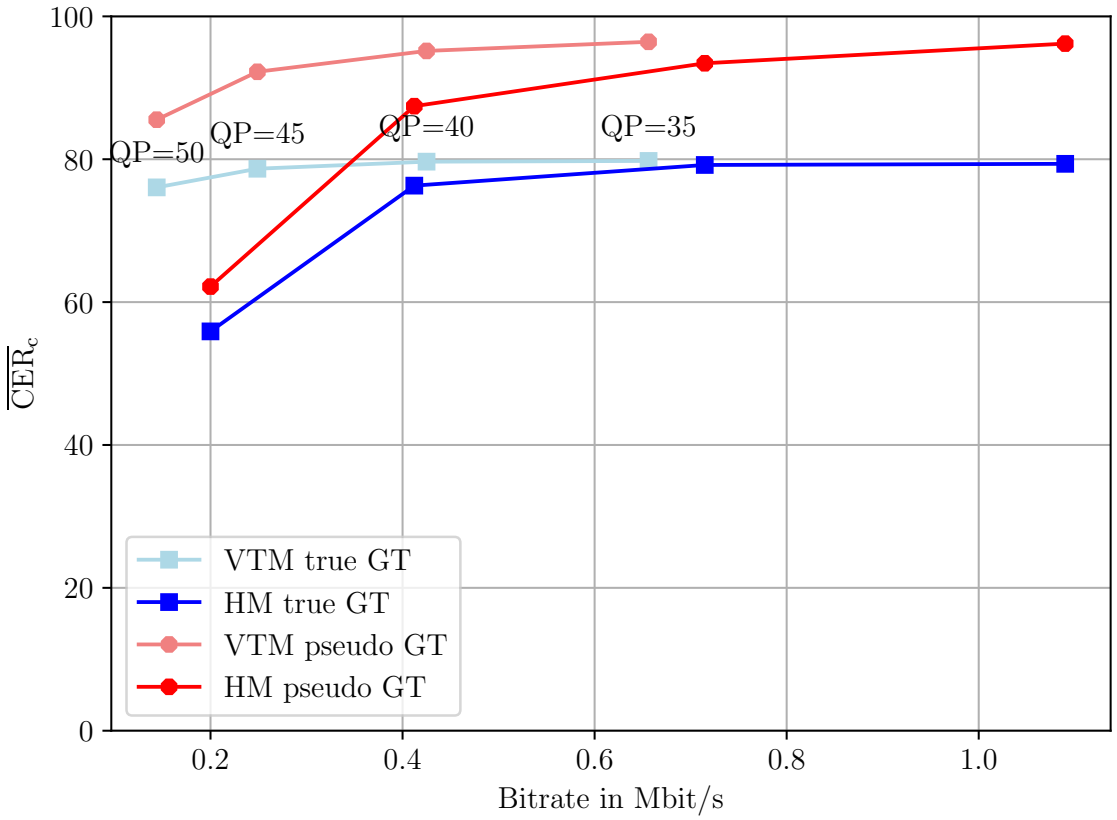


Figure 5.7: Rate-distortion curves for $\overline{\text{CER}}_c$ vs bitrate for selected images encoded with the HM and VTM codec with the default configuration for EasyOCR.

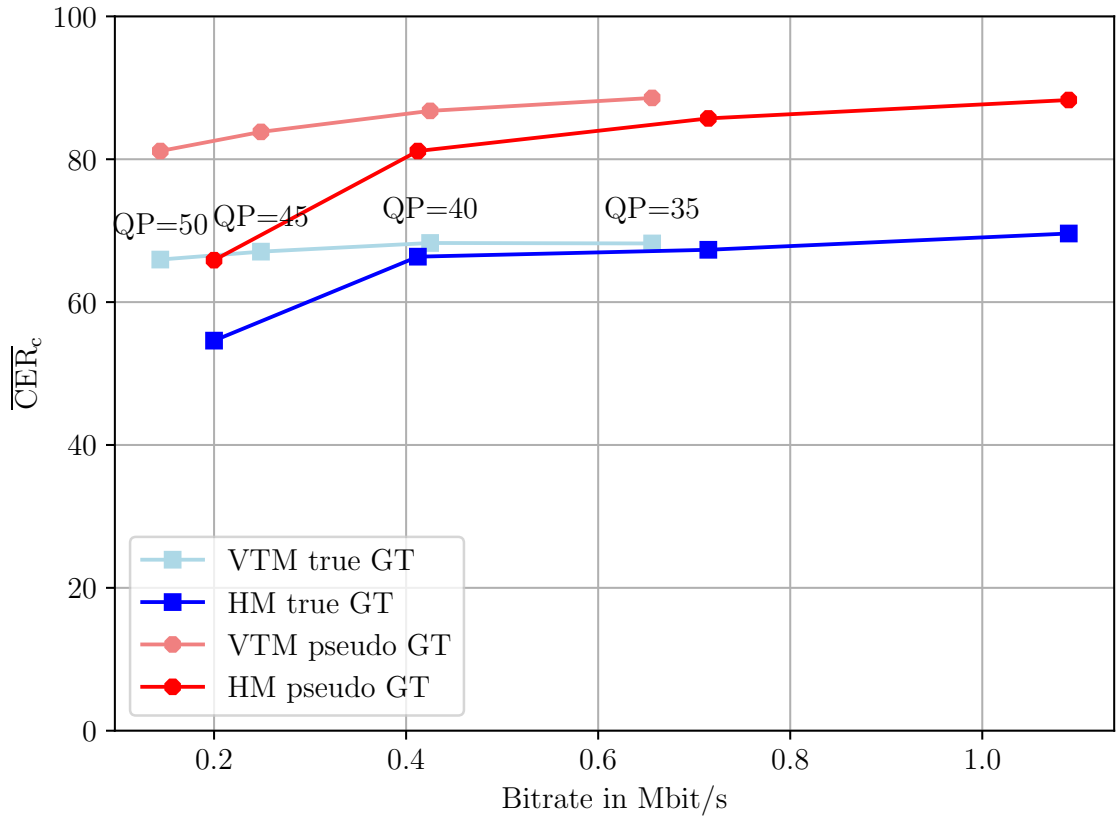


Figure 5.8: Rate-distortion curves for $\overline{\text{CER}}_c$ vs bitrate for selected images encoded with the HM and VTM codec with the default configuration for Tesseract OCR.

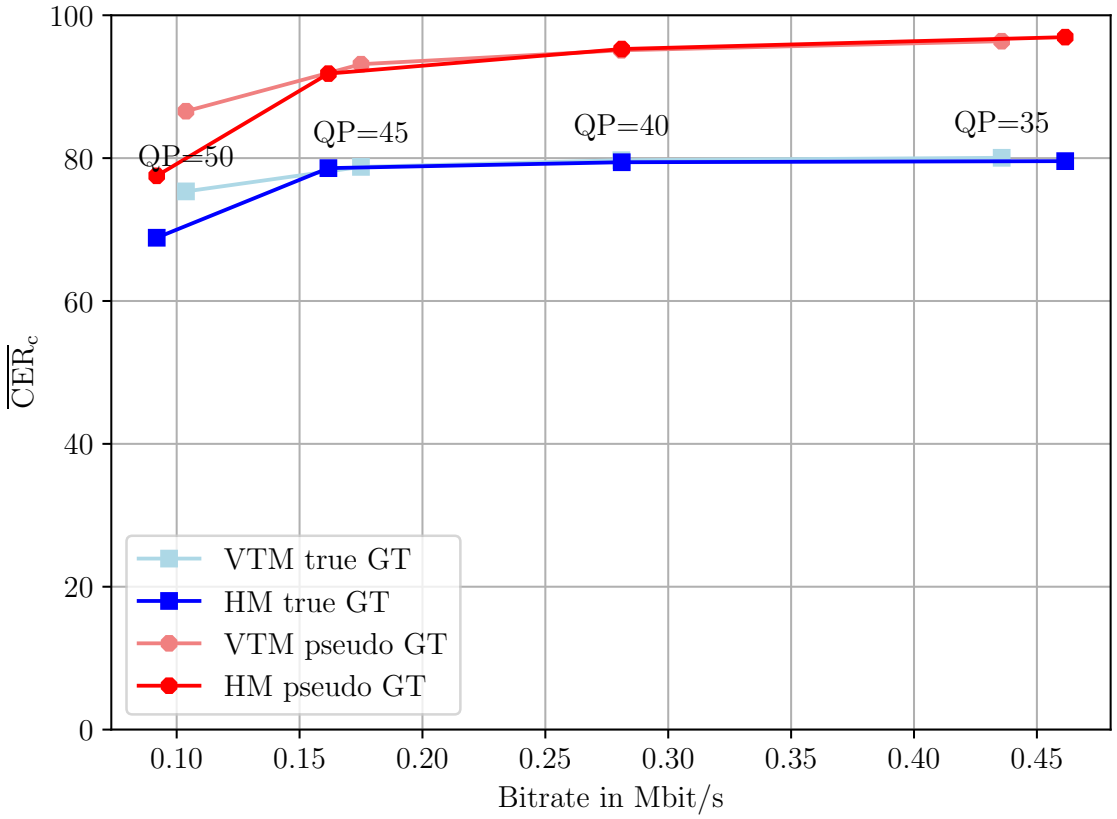


Figure 5.9: Rate-distortion curves for $\overline{\text{CER}}_c$ vs bitrate for selected images encoded with the HM and VTM codec with the SCC configuration for EasyOCR.

default codec configuration for Tesseract OCR. We can observe that the VTM curves are generally further to the top left, compared to the HM curves. One exception is the QP 35, where the VTM codec shows a slightly lower $\overline{\text{CER}}_c$ than the HM codec, but still requires much less bitrate. When comparing the two blue curves with the two red curves, we can see similar trends, like for EasyOCR. However, for the light blue curve the value for QP 35 is lower than for QP 40. In this case we are not able to calculate the BDRate between the curves, since one of the curves is not monotonically increasing, which is an assumption in the BDRate calculation.

In Figure 5.9, we can observe the same curves with the SCC codec configuration for EasyOCR. The SCC configuration enables the HM codec to perform similarly to the VTM codec, with the exception of using a QP of 50, where the VTM performs better. When comparing the blue and red curves, we can see that the trends are very similar. In Figure 5.10, the same curves are depicted with the SCC codec configuration for Tesseract. We can observe the same phenomenon of the HM codec performing similar

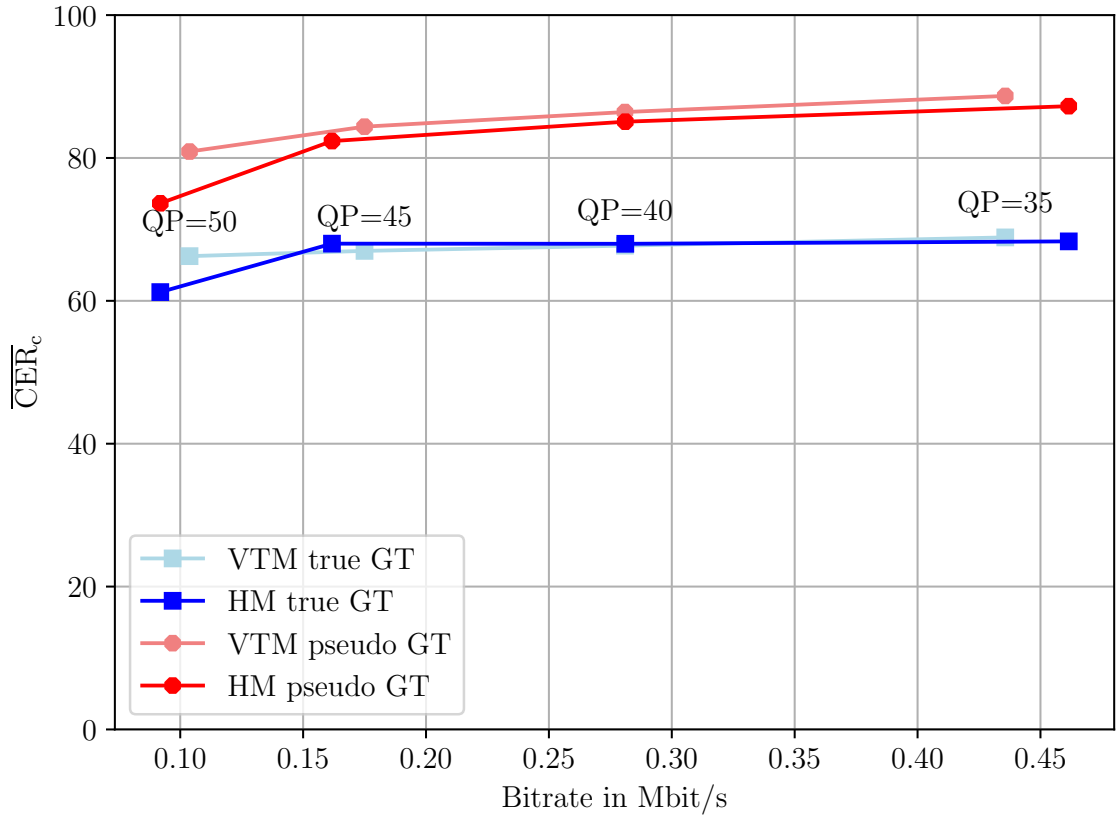


Figure 5.10: Rate-distortion curves for $\overline{\text{CER}}_c$ vs bitrate for selected images encoded with the HM and VTM codec with the SCC configuration for Tesseract OCR.

OCR Algorithm	Codec Configuration	Pseudo GT	True GT	Difference
EasyOCR	Default	-59.28	-61.02	1.73
EasyOCR	SCC	-7.41	-1.1	-6.32
Tesseract	Default	-54.15	—	—
Tesseract	SCC	-25.24	—	—

Table 5.4: Comparison of the BDRate in % between the pseudo and the true GT for the different OCR algorithms and codec configurations.

to the VTM codec, except for a QP of 50, where the VTM outperforms it. When comparing the blue and red curves, we can observe that the trends are not similar. For the true GT, the HM codec performs better than the VTM codec, while for the pseudo GT the VTM codec performs better than the HM codec. To quantify the differences between rate-distortion curves, we calculate the BDRates and compare them.

From Table 5.4, we can see that the BDRate is way higher for the default configuration than for the SCC configuration for the pseudo and the true GT. This reflects the large average distance of the curves seen in Figure 5.7 and Figure 5.8, compared to the slim distance seen in Figure 5.9 and Figure 5.10. For Tesseract OCR, we are not able to calculate the BDRate for the true GT, since the curves are not monotonically increasing. The difference between the pseudo and the true GT is low with 1.73% for the default configuration, but relatively high with -6.31% for the SCC configuration. We can conclude, that for images encoded with the default configurations of the HM and VTM codecs, EasyOCR is able to produce a good pseudo GT.

To summarize, the OCR algorithms perform better on the VTM encoded images than on the HM encoded images. Additionally, the SCC configuration makes the HM codec perform very similarly to the VTM codec. It seems like the SCC configuration manages to keep the text readable even for high QPs values. For the default configuration, EasyOCR seems to be a good pseudo GT. In contrast, Tesseract OCR is most likely not suitable for creating a pseudo GT, as we were not able to calculate the BDRate.

Chapter 6

Conclusion

In this thesis, we investigate the potential of using OCR methods for screen content IQA. We compare the performance of two OCR algorithms, EasyOCR and Tesseract OCR, on the SCID dataset, see section 5.1. First, we conclude, that both OCR methods perform worse on the images affected by MB and GB, with Tesseract OCR performing really poor for GN as well. However, both OCR algorithms perform without impairment for CC, CQD and CSC. Generally, the results show that both OCR methods perform differently for different distortions, but EasyOCR performs better than Tesseract OCR.

Subsequently, we compare the CER_c produced by the OCR methods to human judgment, see section 5.2. We conclude that EasyOCR is generally better suited as a estimation of human judgment compared to Tesseract OCR. For blurred images, EasyOCR exhibits a high correlation with human judgment. Our recommendation is to determine if OCR methods are affected by specific distortions or check which distortions appear in the used data before adding OCR as a metric. Compared to other IQA methods, both OCR algorithms are subpar, especially considering that we selected specifically suited images from the dataset compared to other methods being evaluated on the full dataset. However, our method only uses the text regions of the images, which are missing a lot of information about distortion impacts on the graphical or natural parts of the image. Thus, we recommend combining OCR with other metrics, like the SSIM or the FSIM, to get a more complete picture of the image quality.

Additionally, we surmise that in general EasyOCR performs better than Tesseract OCR on the reference images, but both seem to be too inaccurate to be used as true GT, see section 5.3. Finally, we investigate the performance of the OCR methods for several QPs of the HEVC and VVC codecs and calculate the BDRates between them to compare the true GT with the pseudo GT. We found EasyOCR to be a decent choice for the pseudo GT, especially for the default codec configuration.

For future research, we recommend combining the CER with a metric such as the

intersection over union (IoU) between hand labeled text regions and the prediction bounding boxes. While this may involve significant amount of labeling work, it might lead to a more consistent metric as the order of the text elements becomes less relevant. Thus, it can be used to compare different OCR algorithms more objectively. Moreover, the resulting regions that are not occupied by text elements could be evaluated by other more suitable metrics for pictorial regions, and then combined with the CER into one unified metric. Additionally, using preprocessing steps to alter the performance of the OCR methods and improve the correlation with the MOS might be an interesting research direction.

List of Figures

List of Tables

Bibliography

[1] R. Smith, "An overview of the tesseract ocr engine," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 2, 2007, pp. 629–633. [Online]. Available: https://ieeexplore.ieee.org/document/4376991 ↑1, ↑15

[2] JaidedAI, "Easyocr," https://github.com/JaidedAI/EasyOCR, 2023, [accessed 2023-01-20]. ↑1, ↑14

[3] Z. Ni, L. Ma, H. Zeng, J. Chen, C. Cai, and K.-K. Ma, "ESIM: Edge Similarity for Screen Content Image Quality Assessment," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4818–4831, Oct. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7954714/ ↑1, ↑5, ↑7, ↑21, ↑24, ↑43

[4] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012. [Online]. Available: https://ieeexplore.ieee.org/iel5/76/4358651/06316136.pdf ↑1, ↑28

[5] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (vvc) standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021. [Online]. Available: https://ieeexplore.ieee.org/iel7/76/4358651/09503377.pdf ↑1, ↑29

[6] G. Zhai and X. Min, "Perceptual image quality assessment: a survey," *Science China Information Sciences*, vol. 63, pp. 1–52, 2020. [Online]. Available: https://www.researchgate.net/profile/Guangtao-Zhai/publication/341011181_Perceptual_image_quality_assessment_a_survey/links/61812d24a767a03c14e3d754/Perceptual-image-quality-assessment-a-survey.pdf ↑3, ↑6, ↑21

[7] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2366–

2369. [Online]. Available: https://projet.liris.cnrs.fr/imagine/pub/proceedings/ICPR-2010/data/4109c366.pdf ↑4

[8] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004. [Online]. Available: https://ece.uwaterloo.ca/~z70wang/publications/ssim.pdf ↑4

[9] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "A comprehensive evaluation of full reference image quality assessment algorithms," *2012 19th IEEE International Conference on Image Processing*, pp. 1477–1480, 2012. [Online]. Available: http://www4.comp.polyu.edu.hk/~cslzhang/paper/conf/ICIP12.pdf ↑4

[10] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2. Ieee, 2003, pp. 1398–1402. [Online]. Available: https://utw10503.utweb.utexas.edu/publications/2003/zw_asil2003_msssim.pdf ↑4

[11] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "Fsim: A feature similarity index for image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011. [Online]. Available: https://ieeexplore.ieee.org/document/5705575 ↑4, ↑7

[12] H. Yang, Y. Fang, and W. Lin, "Perceptual quality assessment of screen content images," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 4408–4421, 2015. [Online]. Available: http://sim.jxufe.cn/JDMKL/pdf/Perceptual%20quality%20assessment%20of%20screen%20content%20images.pdf ↑4, ↑5

[13] Z. Ni, L. Ma, H. Zeng, Y. Fu, L. Xing, and K.-K. Ma, "Scid: A database for screen content images quality assessment," in *2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, 2017, pp. 774–779. [Online]. Available: https://eezkni.github.io/publications/conference/SCID/SCID_ZKNI_ISPACS17.pdf ↑4

[14] L. Kang, P. Ye, Y. Li, and D. Doermann, "A deep learning approach to document image quality assessment," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 2570–2574. [Online]. Available: https://projet.liris.cnrs.fr/imagine/pub/proceedings/ICIP-2014/Papers/1569912503.pdf ↑5

[15] Y. Fang, J. Yan, J. Liu, S. Wang, Q. Li, and Z. Guo, "Objective quality assessment of screen content images by uncertainty weighting," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 2016–2027, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7857067 ↑5

[16] Z. Ni, H. Zeng, L. Ma, J. Hou, J. Chen, and K.-K. Ma, "A gabor feature-based quality assessment model for the screen content images," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4516–4528, 2018. [Online]. Available: https://forestlinma.com/welcome_files/zkni_TIP_2018.pdf ↑5

[17] V. Q. E. G. (VQEG), "Final report from the video quality experts group on the validation of objective models of video quality assessment," 2003. [Online]. Available: https://vqeg.org/VQEGSharedFiles/Publications/Validation_Tests/FRTV_Phase2/FRTV_Phase2_Final_Report.pdf ↑6, ↑7

[18] A. Rohaly, P. Corriveau, J. Libert, A. Webster, V. Baroncini, J. Beerends, J.-L. Blin, L. Contin, T. Hamada, D. Harrison, A. Hekstra, J. Lubin, Y. Nishida, R. Nishihara, J. Pearson, A. Pessoa, N. Pickford, A. Schertz, M. Visca, and S. Winkler, "Video quality experts group: current results and future directions," vol. 4067, 05 2000, pp. 742–753. [Online]. Available: https://www.researchgate.net/publication/221458323_Video_Quality_Experts_Group_current_results_and_future_directions ↑6

[19] D. Li, T. Jiang, and M. Jiang, "Exploiting high-level semantics for no-reference image quality assessment of realistic blur images," in *Proceedings of the 25th ACM international conference on Multimedia*. ACM, Oct. 2017. [Online]. Available: https://arxiv.org/pdf/1810.08169.pdf ↑6, ↑7

[20] Y. Fang, J. Yan, J. Liu, S. Wang, Q. Li, and Z. Guo, "Objective quality assessment of screen content images by uncertainty weighting," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 2016–2027, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7857067 ↑7

[21] P. Mohammadi, A. Ebrahimi-Moghadam, and S. Shirani, "Subjective and objective quality assessment of image: A survey," 2014. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1406/1406.7799.pdf ↑7

[22] S.-C. Pei and L.-H. Chen, "Image quality assessment using human visual dog model fused with random forest," *IEEE Transactions on Image*

Processing, vol. 24, no. 11, pp. 3282–3292, 2015. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7115917 ↑7

[23] A. Alaei, V. Bui, D. Doermann, and U. Pal, "Document image quality assessment: A survey," *ACM Computing Surveys*, 2023. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/3606692 ↑7

[24] J. Ruikar and S. Chaudhury, "Nits-iqa database: A new image quality assessment database," *Sensors*, vol. 23, no. 4, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/4/2279 ↑7

[25] M. Oszust, "Full-reference image quality assessment with linear combination of genetically selected quality measures," *PloS one*, vol. 11, no. 6, p. e0158333, 2016. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4920377/ ↑7

[26] H. Sheikh, M. Sabir, and A. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006. [Online]. Available: https://live.ece.utexas.edu/publications/2006/hrs-transIP-06.pdf ↑7

[27] W. Sun, F. Zhou, and Q. Liao, "Mdid: A multiply distorted image database for image quality assessment," *Pattern Recognition*, vol. 61, pp. 153–168, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0031320316301911?via%3Dihub ↑7

[28] J. J. Moré, "The levenberg-marquardt algorithm: Implementation and theory," in *Numerical Analysis*, G. A. Watson, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1978, pp. 105–116. [Online]. Available: https://www.osti.gov/servlets/purl/7256021 ↑7

[29] L. Wang, "A survey on iqa," 2022. [Online]. Available: https://arxiv.org/pdf/2109.00347.pdf ↑7

[30] J. de Winter, S. Gosling, and J. Potter, "Comparing the pearson and spearman correlation coefficients across distributions and sample sizes: A tutorial using simulations and empirical data," *Psychological Methods*, vol. 21, pp. 273–290, Sep. 2016. [Online]. Available: https://gwern.net/doc/statistics/order/2016-dewinter.pdf ↑7, ↑8

[31] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *ITU SG16 Doc. VCEG-M33*, 2001. ↑9

[32] C. Herglotz, M. Kränzler, R. Mons, and A. Kaup, "Beyond bjøntegaard: Limits of video compression performance comparisons," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 46–50. [Online]. Available: https://arxiv.org/pdf/2202.12565.pdf ↑9, ↑10, ↑53

[33] N. Islam, Z. Islam, and N. Noor, "A survey on optical character recognition system," 2017. [Online]. Available: https://arxiv.org/abs/1710.05703 ↑13

[34] A. Asif, S. A. Hannan, Y. Perwej, and M. A. Vithalrao, "An overview and applications of optical character recognition," *Int. J. Adv. Res. Sci. Eng*, vol. 3, no. 7, pp. 261–274, 2014. ↑13

[35] K. Hamad and K. Mehmet, "A detailed analysis of optical character recognition technology," *International Journal of Applied Mathematics Electronics and Computers*, no. Special Issue-1, pp. 244–249, 2016. [Online]. Available: https://dergipark.org.tr/en/download/article-file/236939 ↑13

[36] S. Singh, "Optical character recognition techniques: a survey," *Journal of emerging Trends in Computing and information Sciences*, vol. 4, no. 6, 2013. [Online]. Available: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=375e6f94bb9039f3df4fa2a625f11ac59db3629f ↑13

[37] C. Wick, C. Reul, and F. Puppe, "Calamari-a high-performance tensorflow-based deep learning package for optical character recognition," *arXiv preprint arXiv:1807.02004*, 2018. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1807/1807.02004.pdf ↑14

[38] T. C. Wei, U. Sheikh, and A. A.-H. Ab Rahman, "Improved optical character recognition with deep neural network," in *2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA)*. IEEE, 2018, pp. 245–249. [Online]. Available: https://ieeexplore.ieee.org/document/8368720 ↑14

[39] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *CoRR*, vol. abs/1507.05717, 2015. [Online]. Available: http://arxiv.org/abs/1507.05717 ↑14, ↑53

[40] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9365–9374. [Online]. Available: https://arxiv.org/pdf/1904.01941.pdf ↑14

[41] S. Weil, R. Smith, and P. Zdenko, "Tesseract ocr," https://github.com/tesseract-ocr/tesseract, 2023, [accessed 2023-01-20]. ↑15

[42] R. Smith, "Tesseract blends old and new ocr technology," https://tesseract-ocr.github.io/docs/das_tutorial2016/2ArchitectureAndDataStructures.pdf, 2016, [accessed 2023-07-20]. ↑15

[43] S. Hoffstaetter, J. Bochi, M. Lee, L. Kistner, R. Mitchell, E. Cecchini, J. Hagen, D. Morawiec, E. Bedada, and U. Akyüz, "Pytesseract," https://github.com/madmaze/pytesseract, 2022, [accessed 2023-01-20]. ↑16

[44] D. Vedhaviyassh, R. Sudhan, G. Saranya, M. Safa, and D. Arun, "Comparative analysis of easyocr and tesseractocr for automatic license plate recognition using deep learning algorithm," in *2022 6th International Conference on Electronics, Communication and Aerospace Technology*. IEEE, 2022, pp. 966–971. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10009215/ ↑16

[45] T. Singkhornart and O. Surinta, "Multi-language video subtitle recognition with convolutional neural network and long short-term memory networks," *ICIC Express Letters*, vol. 16, pp. 647–655, 06 2022. [Online]. Available: http://www.icicel.org/ell/contents/2022/6/el-16-06-10.pdf ↑18

[46] J. Xu, R. Joshi, and R. A. Cohen, "Overview of the emerging hevc screen content coding extension," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 50–62, 2015. [Online]. Available: https://merl.com/publications/docs/TR2015-126.pdf ↑29

[47] Joint Collaborative Team on Video Coding. HEVC test model reference software (HM). https://hevc.hhi.fraunhofer.de/. [accessed 2023-02-20]. ↑29

[48] F. Bossen, "Jctvc-l1100: Common test conditions and software reference configurations," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, Tech. Rep., Jan. 2013. ↑29

[49] ——, "Jctvc-u1015-r2: Common test conditions for screen content coding," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Warsaw, Poland, Tech. Rep., June 2015. ↑29

[50] Joint Video Experts Team (JVET). VVC test model reference software (VTM). https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM. [accessed 2023-02-20]. ↑29

[51] Y. Chao, Y. Sun, J. Xu, and X. Xu, "JVET common test conditions and software reference configurations for non-4:2:0 colour formats," Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, AHG Report, JVET-R2013. ↑29

[52] T. Tang, L. Li, X. Wu, R. Chen, H. Li, G. Lu, and L. Cheng, "Tsa-scc: Text semantic-aware screen content coding with ultra low bitrate," *IEEE Transactions on Image Processing*, vol. 31, pp. 2463–2477, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9720120 ↑31

[53] K. Gu, S. Wang, G. Zhai, S. Ma, and W. Lin, "Screen image quality assessment incorporating structural degradation measurement," vol. 2015, pp. 125–128, 07 2015. [Online]. Available: https://www.researchgate.net/publication/285429613_Screen_image_quality_assessment_incorporating_structural_degradation_measurement ↑43

Curriculum Vitae

Personal Data

Address	Moltkestraße 6 91054 Erlangen
Email	sebastian.hirt@fau.de
Date of birth	26.07.1996
Nationality	german

Education

09/2006 - 07/2014	Werner-von-Siemens-Gymnasium, Weißenburg
10/2014 - 09/2020	Bachelor of Science Friedrich-Alexander-Universität Erlangen-Nürnberg
10/2020 - 09/2023	Master of Science Friedrich-Alexander-Universität Erlangen-Nürnberg

Erlangen, August 15, 2023

Sebastian Hirt