Informatics

Applications of Neural Attention for Modelling Long-Range Dependencies

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Technischen Wissenschaften

by

Dipl.-Ing. Dipl.-Ing. Matthias Gerold Wödlinger
Registration Number 01125717

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Robert Sablatnig

The dissertation has been reviewed by:

Prof. Dr.-Ing. habil. Andreas Maier

Prof. Dr. rer. nat. habil. Timo Ropinski

Vienna, 26th August, 2024

Matthias Gerold Wödlinger

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Dipl.-Ing. Dipl.-Ing. Matthias Gerold Wödlinger

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 26. August 2024

Matthias Gerold Wödlinger

Acknowledgements

First, I would like to express my sincere gratitude to my thesis advisor, Robert, for his support and understanding and for creating a work environment that is both stimulating for research ideas and fun during coffee breaks and after-work events. All of this greatly contributed to my graduate experience. I am also grateful for having had the opportunity to work on several exciting projects during my studies.

My thanks also go to Katharina, our secretary, who not only helped me manage bureaucracy but also was my sole support in drinking Radler instead of beer.

I would like to thank my colleagues for interesting discussions, fun parties, exciting Mario Kart tournaments, and an overall great time.

I also want to thank my family, who have always been there for me, no matter the circumstances. I am grateful for the support and environment you have always provided, allowing me to pursue my passions.

Finally, I want to thank my wife, Ghazal, who supported me in every decision of this journey and made the multiple lockdowns during COVID-19 enjoyable.

Thank you all!

Danksagung

Zunächst möchte ich mich bei meinem Betreuer Robert Bedanken für seine Unterstützung, sein Verständnis und die Schaffung eines Arbeitsumfelds, das sowohl anregend für Forschungsideen als auch unterhaltsam während Kaffeepausen und diversen Veranstaltungen nach (während) der Arbeit ist. All dies hat sehr zu meiner positiven Erfahrung als Doktorand beigetragen. Ich bin auch dankbar dafür, dass ich die Gelegenheit hatte, während an mehreren interessanten Projekten zu arbeiten.

Mein Dank geht auch an Katharina, unsere Sekretärin, die mir nicht nur bei der Bewältigung der Bürokratie geholfen hat, sondern auch meine einzige Unterstützung beim Trinken von Radler statt Bier war.

Ich möchte mich bei all meinen Kollegen für interessante Diskussionen, lustige Partys, spannende Mario-Kart-Turniere und insgesamt eine tolle Zeit bedanken.

Ich möchte mich auch bei meiner Familie bedanken, die immer für mich da war, egal unter welchen Umständen. Ich bin dankbar für die Unterstützung und das Umfeld, das sie mir immer geboten haben und das es mir ermöglicht hat, meinen Leidenschaften nachzugehen.

Schließlich möchte ich bei meiner Frau Ghazal bedanken, die mich bei jeder Entscheidung auf dieser Reise unterstützt hat und mir selbst die zahlreichen Lockdowns während Covid ertragbar gemacht hat.

Danke!

Abstract

Dominant neural network architectures such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are inherently local and require deep networks to incorporate distant information. Early work on visual attention allowed the modelling of non-local structures through sequences of local glimpses, and more recently, the Transformer family of networks revolutionised neural attention by using self-attention to process all inputs in parallel.

This thesis explores applications of neural attention mechanisms in computer vision, focusing on modelling long-range dependencies in three domains: historical document analysis, biomedical data processing, and stereo-image compression. Moreover, it introduces new methods that use attention mechanisms to address specific challenges in these domains.

A visual attention-based method for extracting text baselines from images of historical handwritten texts is presented. The proposed method relies on a network that sequentially shifts attention along text lines to determine polygon coordinates. The method allows for direct learned prediction of text baseline coordinates rather than relying on heuristics.

In the biomedical domain, the thesis introduces the Flowformer model, an efficient variant of the Transformer architecture designed to process large flow cytometry samples for cancer cell detection. The proposed approach differs from traditional methods by using global attention to model samples holistically. As a result, it achieves state-of-the-art performance in cancer cell identification across multiple datasets.

Finally, two stereo image compression models, SASIC and ECSIC, are presented, which use cross-attention mechanisms to model the mutual information between stereo image pairs. These methods achieve state-of-the-art compression performance by efficiently capturing redundancies between images while maintaining fast runtimes.

This thesis provides explicit solutions to real-world problems in document analysis, biomedical data processing, and image compression. It also demonstrates the effectiveness of neural attention in dealing with different long-range dependencies.

Kurzfassung

Populäre neuronale Netzwerkarchitekturen wie Convolutional Neural Networks (CNNs) oder Recurrent Neural Networks (RNNs) sind per Konstruktion lokal und benötigen üblicherweise tiefe Netzwerke, um nicht-lokale Abhängigkeiten zu modellieren. Frühe Arbeiten zur Neural Attention ermöglichten die Modellierung von nicht-lokalen Strukturen durch Sequenzen von lokalen *Glimpses*. In 2017 revolutionierte die Transformer-Netzwerkfamilie Neural Attention, indem sämtliche Positionen und Features eines Samples parallel verarbeitet wurden, durch die Verwendung von Self-Attention.

Diese Dissertation erforscht Anwendungen von Methoden der Neural Attention in der Computer Vision. Der Fokus liegt dabei auf der Modellierung von nicht-lokalen Abhängigkeiten innerhalb von Daten in drei verschiedenen Anwendungsbereichen: Historische Dokumentenanalyse, biomedizinische Daten verarbeitung und Stereo-Bildkompression. Es werden neue Methoden vorgestellt, die Neural Attention verwenden, um spezifische Herausforderungen in diesen Bereichen zu bewältigen.

Es wird eine Methode vorgestellt, die auf Visual Attention basiert und zur Extraktion von Textzeilen aus Bildern historischer Manuskripte dient. Die vorgeschlagene Methode basiert auf einem Netzwerk, das Bilder von Dokumenten sequentiell verarbeitet, um Textlinien als Polygonkoordinaten zu bestimmen. Die Methode ermöglicht eine direkte, erlernte Vorhersage von Textlinienkoordinaten ohne Rückgriff auf Extraktionsheuristiken.

Im biomedizinischen Bereich wird in der Dissertation das Flowformer-Modell vorgestellt. Es handelt sich dabei um eine effiziente Variante der Transformer-Architektur, die in der Lage ist, selbst große Flowzytometrie-Proben für die Erkennung von Krebszellen zu verarbeiten. Das Modell unterscheidet sich von traditionellen Methoden durch die Verwendung von Global Attention, um Proben ganzheitlich zu modellieren. Die Methode erreicht State-of-the-Art-Leistung bei der Identifizierung von Krebszellen über mehrere Datensätze.

Abschließend werden zwei Stereo-Bildkompressionsmodelle vorgestellt: SASIC und ECSIC. Beide Modelle verwenden Cross-Attention, um die gemeinsame Information zwischen Stereobildpaaren zu modellieren. Diese Methoden erreichen State-of-the-Art Kompressionsleistung, indem sie Redundanzen zwischen Bildern erfassen und gleichzeitig die Laufzeiten minimal halten.

In dieser Dissertation werden explizite Lösungen für reale Probleme in der Dokumentenanalyse, der biomedizinischen Datenverarbeitung und der Bildkompression vorgestellt. Weiter zeigt diese Arbeit die Wirksamkeit von Neural Attention bei der Lösung von Anwendungen, die nicht-lokale Abhängigkeiten erfordern.

Contents

Abstract ix

Kurzfassung xi

Contents xiii

1 Introduction 1
1.1 Motivation . . . 1
1.2 Overview of Contributions . . . 4
1.3 Thesis Structure . . . 6

2 Related Work 9
2.1 Early Work . . . 9
2.2 Visual Attention . . . 12
2.3 Transformers . . . 13
2.4 Transformers for Vision . . . 16
2.5 Efficient Transformers . . . 19
2.6 Summary . . . 21

3 Methodology 23
3.1 Modelling Text Baselines with Visual Attention . . . 23
3.1.1 Related Work . . . 24
3.1.2 Methodology . . . 26
3.1.3 Evaluation . . . 30
3.1.4 Conclusion . . . 33
3.2 Modelling Flow Cytometry Data with Global Attention . . . 35
3.2.1 Related Work . . . 36
3.2.2 Methodology . . . 37
3.2.3 Evaluation . . . 40
3.2.4 Conclusion . . . 45
3.3 Modelling Stereo Image Redundancies with Stereo Cross Attention . . . 46
3.3.1 Related Work . . . 47
3.3.2 Methodology . . . 49
3.3.3 Evaluation . . . 59

3.3.4 Conclusion . . . 65
3.4 Summary . . . 67

4 Conclusion 69

A Additional Results for Flowformer++ 75
A.1 Ablation study: Flowformer vs Flowformer++ . . . 75
A.2 Calibration . . . 76

B Qualitative Stereo Image Compression Results 91
B.1 SASIC . . . 91
B.2 ECSIC . . . 93

Bibliography 97

CHAPTER 1

Introduction

Neural attention describes several techniques inspired by the attention mechanism in the human brain. These methods work by allowing neural networks to attend to different locations in the data, either sequentially or in parallel. This property enables these networks to learn non-local structures and sequences. This thesis discusses problems in three domains (historical document images, flow cytometry data, and stereo images), each with unique challenges, and proposes attention-based methods for each.

1.1 Motivation

Since 2012, when AlexNet [KSH12] won the ImageNet [DDS+09] competition, deep neural networks have proven to be a capable method for various applications. In computer vision, Convolutional Neural Networks (CNNs) are typically used to extract features learned from a large dataset. For sequential data, recurrent neural networks and their derivatives, such as Long Short-Term Memory cells (LSTMs) [HS97] and Gated Recurrent Units (GRUs) [WTD+18] have similarly outperformed other methods at the time for tasks like machine translation [BCB15] or speech recognition [GMH13].

However, these methods, by design, process data locally. While local features have been shown to work well for a wide variety of tasks, some problems might benefit from a more holistic view, and others require features that are inherently non-local and span over long distances. CNNs consist of layers that are collections of learnable local filters. Every layer can only extract local features from the preceding layer's output. Only by stacking multiple layers is it possible to extract features that can model non-local dependencies. However, this requires deep networks. For example, a stack of 10 layers of 3 × 3 convolutions with a stride of 1 has a receptive field of 21 × 21. To achieve receptive fields that span entire images, deep networks are required, along with stratified convolutions and aggressive pooling. This leads to problems like loss of information due to pooling and increased resource requirements due to deeper networks. Recurrent neural networks are, in theory, capable of storing information for arbitrary long distances. However, in practice, the gradients corresponding to long-range dependencies often either blow up or vanish [HS97]. A remedy for this was proposed in Hochreiter et al. [HS97] with the Long Short-Term Memory cell. However, their sequential design prevents them from being trained in parallel.

To address these challenges, Vaswani et al. [VSP+17a] proposed the Transformer network. It is a feed-forward network that processes a fixed-length sequence of embedding vectors in a single forward pass. The network comprises a sequence of Transformer layers that have the self-attention operation at their core[1]:

$$\mathrm{SelfAttn}(X) = \mathrm{softmax}\left(\frac{XW_Q W_K^\intercal X^\intercal}{\sqrt{d}}\right) XW_V. \tag{1.1}$$

Here, $X \in \mathbb{R}^{N\times m}$ represents the N input vectors. The matrices $W_Q \in \mathbb{R}^{m\times d}$, $W_K \in \mathbb{R}^{m\times d}$ and $W_V \in \mathbb{R}^{d\times d}$ indicate the linear embedding maps and d denotes the dimensionality of the embedding space. Notably, this operation is permutation equivariant (permutations of the input sequence, i.e. permutations in the dimension N of X). Thus, typical sequence applications where the input order has some relevance require additional positional embedding [VSP+17a] of the input.

The attention mechanism draws inspiration from the human brain's ability to selectively focus on particular aspects of the environment. This selective view of data improves the capabilities of a network to handle tasks involving complex data structures and relationships, such as the sequential and spatial dependencies found in natural language and visual scenes, by emphasizing relevant information while disregarding extraneous details.

Unlike CNNs or RNNs, attention allows the modelling of arbitrarily long-range dependencies in a single layer. This can be seen from eq. (1.1) by observing that every position of X attends (i.e. a dot product is performed to compare the embedding vectors at two positions) to every other position in X. However, the size of the resulting matrix

$$XW_Q W_K^\intercal X^\intercal =: \mathrm{AttnMatrix} \in \mathbb{R}^{N\times N} \tag{1.2}$$

grows quadratically $\mathcal{O}(N^2)$ in the sequence length. For this reason, the sequence was originally restricted to a local neighbourhood [VSP+17a]. This prevents the method from being directly applied to longer sequences without additional considerations. Recently, there have been a plethora of works that address these issues [TDA+20, TDBM20]. However, these methods typically focus on problems from natural language processing. Only recently, variants of the Transformer have been proposed that work on images [DBK+20, LLC+21], and these methods process images in a patch-based manner. Since these initial patches are projected to embedding vectors with a simple linear layer, the architecture can struggle with smaller local structures.

[1] The self-attention blocks of the original Transformer also contain residual connections, element-wise linearities and layer normalization.

Accurate modelling of non-local dependencies is necessary for various applications, such as understanding long text documents in NLP, modelling long-range dependencies between objects at different locations in images, and modelling holistic information in data. While the problem of modelling long text documents has received much interest from academia and industry due to recent developments in Large Language Models (LLMs) for many more niche applications, such as stereo image data or specific medical data types, the problem of modelling such non-local structures with neural attention is still underexplored.

This thesis investigates the problem of modelling long-range dependencies by focusing on three variants of neural attention. For each variant, novel methods for a specific application where attention has the potential to solve previously open problems are developed.

1. **Visual Attention for predicting linear sequences in images:** Text baselines in historical handwritten documents are notoriously difficult to identify [DKF+17]. The ink in these documents is often degraded, shows bleed-through from text written on the back and has a high variation in writing styles and languages. Detecting the text baselines is a necessary step in current methods for handwritten text recognition. These methods are typically based on sequences of windows, which rely on extracted text baselines. Current methods aim to detect text baselines, typically with segmentation methods followed by heuristic rules to extract a polygon [GLS+19]. However, a reliable segmentation requires large amounts of training data of documents with a similar structure to the test data. Furthermore, because the full document page needs to be processed at once, the GPU memory is a natural bottleneck for the maximal document resolution. This thesis investigates how the sequential nature of these baselines can be exploited to extract baselines in a recurrent manner using visual attention techniques.

2. **Efficient Self-Attention to capture long-range information in large samples:** Acute leukaemia is the most frequent cancer in children and adolescents. Identifying cancer cells and assessing the treatment response is typically performed based on cell marker measurements of a bone mark sample. These samples are processed by Flow Cytometer Machines (FCM) that measure the surface proteins and return a vector of marker intensities (i.e. for a sample containing n cells with m markers being measured, an FCM machine returns a high dimensional point cloud $\in \mathbb{R}^{n \times m}$). These samples typically contain $10^5 - 10^6$ cells. The structure of this data type prevents naive applications of existing methods like CNNs or RNNs. Current methods for blast detection in FCM data either work by estimating the density with traditional parametric methods like Gaussian Mixture Methods [RDS+19] that are unable to model complicated structures due to their inherent bias and are computationally expensive during inference or by ignoring long-range dependencies and processing different cells independently of each other [LSR+18] which results in fixed decision boundaries. These models have difficulty coping

with variations in the population distribution in samples from different sources (e.g. different clinical centres, different operators, different FCM machines) and different patients. Correct identification of cancer cells requires a holistic view of the whole sample [RDS+19], which motivates the work conducted in this thesis. This thesis investigates how to learn the global structure of FCM files and use it to accurately detect blast cells.

3. **Modelling mutual information in stereo image pairs with cross attention:** Stereo images consist of a pair of images from the same scene taken from different locations. Due to the resulting overlap in the field of view, these image pairs have substantial mutual information $I(l, r)$. Conventional compression of stereo images treats both images in a pair as separate, ignoring any shared information and therefore leading to suboptimal compression bitrates $R(l) + R(r) \leq R(l) + R(r) - I(l, r)$. A method capable of modelling this shared information can substantially improve compression performance.

   Some methods attempt to do this with heuristic matching methods [MMSW06]. Others employ 3D convolutions to process a disparity cost volume [LWU19b] or homographic warping [DYY+21]. However, these methods are slow or only able to model linear warpings. In recent years, learned methods for compression have outperformed traditional methods in PSNR [BLS17] and perceptual metrics [MTTA20]. These methods typically follow the structure proposed by Ballé et al. [BLS17] of an encoder-decoder network, trained end-to-end with a rate-distortion loss, optionally with the inclusion of side information via an additional encoder-decoder network [BMS+18] or autoregressive components [MBT18]. These models are typically fully-convolutional which might lead to sub-optimal performance because convolutions apply a fixed set of filters independent of the input (i.e. they are not data-specific), and they have been shown to act as high pass filters [PK22] amplifying the noisy (i.e. high entropy but low importance) aspects of images. Attention has been shown to counteract some of these issues [PK22], and its data specificity can help model sample-dependent disparity distributions that, in some instances like outdoor scenes, require accurate modelling of non-local connections between both images in a stereo image pair.

1.2 Overview of Contributions

The field of neural attention in vision has changed while writing this thesis. While initial work on attention in vision followed the structure of the seminal work from Xu et al. [XBK+15] with an RNN guiding the attention of a CNN that performs the actual prediction (typically called *visual attention*), nowadays attention is understood synonymously with the Transformer architecture [VSP+17a]. The contributions in this thesis follow a clear progression of research. The initial work targets visual attention, while the remainder of the thesis mainly focuses on variants and applications of the attention mechanism employed in the Transformer family of models. Each contribution targets applications in one of the three domains: Historical document images, flow cytometry data, and stereo images. The domains investigated are sufficiently different to allow the investigation of various aspects of attention and other meanings of long-range.

Visual Attention for predicting linear sequences in images This work presents a novel method that extracts text baselines without any heuristics, relying solely on a visual attention network that "follows" the text line from start to end and extracts the line polygon coordinates. This work is published in the proceedings of the International Conference for Pattern Recognition 2020 (ICPR) [WS21].

Efficient Self-Attention to capture holistic information in large samples This work presents a novel method that processes full FCM samples using an efficient Transformer variant. The model performs global attention and predicts binary labels to differentiate cancer cells from healthy cells. Global attention allows the model to extract holistic features that lead to predictions that are robust with respect to population changes due to operator or clinical laboratory differences. It is the first neural network that achieves state-of-the-art performance on multiple public datasets while being fast enough to process a complete sample in less than a second. This work is published in the Journal for Computers in Biology and Medicine [WRW$^+$22b].

Modelling mutual information in stereo image pairs with cross attention In this work, two methods are proposed for the compression of stereo image pairs. The first model, SASIC [WKXS22], combines heuristic warping with a decoder that performs stereo attention. It is published in the proceedings of the Conference for Computer Vision and Pattern Recognition 2022 (CVPR). The second method, ECSIC, works without any heuristics and instead models the mutual information between the images of a stereo image pair with the stereo cross-attention module and two stereo context modules. It is published in the proceedings of the Winter Conference on Applications of Computer Vision 2024 (WACV) [WKK$^+$24]. Both methods achieved state-of-the-art performance on popular open stereo image datasets at the time of publication.

Finally, the following publications were part of my research but are not covered in this thesis:

- Matthias Wödlinger and Robert Sablatnig. Classification and segmentation of scanned library catalogue cards using convolutional neural networks. In *Proceedings of the Joint Austrian Computer Vision and Robotics Workshop 2020*, pages 90–91. Austrian Association for Pattern Recognition (ÖAGM/AAPR), 2020

- Lisa Weijler, Florian Kowarsch, Matthias Wödlinger, Michael Reiter, Margarita Maurer-Granofszky, Angela Schumich, and Michael N Dworzak. Umap based anomaly detection for minimal residual disease quantification within acute myeloid leukemia. *Cancers*, 14(4):898, 2022

- Florian Kowarsch, Lisa Weijler, Matthias Wödlinger, Michael Reiter, Margarita Maurer-Granofszky, Angela Schumich, Elisa O Sajaroff, Stefanie Groeneveld-Krentz, Jorge G Rossi, Leonid Karawajew, et al. Towards self-explainable transformers for cell classification in flow cytometry data. In *International Workshop on Interpretability of Machine Intelligence in Medical Image Computing*, pages 22–32. Springer, 2022

- Thomas Heitzinger, Matthias Woedlinger, and David G Stork. Artist-specific style transfer for semantic segmentation of paintings: The value of large corpora of surrogate artworks. *Electronic Imaging*, 34(13):186–1, 2022

- Alexander Bayerl, Manuel Keglevic, Matthias Wödlinger, and Robert Sablatnig. Impact of learned domain specific compression on satellite image object classification. In *26th Computer Vision Winter Workshop (CVWW)*, 2023

- Jan Kotera, Matthias Wödlinger, and Manuel Keglevic. Learned lossy image compression for volumetric medical data. In *26th Computer Vision Winter Workshop (CVWW)*, 2023

- Florian Kowarsch, Lisa Magdalena Weijler, Matthias Gerold Wödlinger, Florian Kleber, Margarita Maurer-Granofszky, Michael Reiter, and Michael Dworzak. Explainable visualization techniques for transformers in flow cytometry data. [Conference Presentation]. 26th Computer Vision Winter Workshop (CVWW), 2023

- Florian Kowarsch, Margarita Maurer-Granofszky, Lisa Weijler, Matthias Wödlinger, Michael Reiter, Angela Schumich, Tamar Feuerstein, Simona Sala, Michaela Nováková, Giovanni Faggin, et al. Fcm marker importance for mrd assessment in t-cell acute lymphoblastic leukemia: An aieop-bfm-all-flow study group report. *Cytometry Part A*, 105(1):24–35, 2024

1.3 Thesis Structure

The remainder of this thesis is structured as follows. Chapter 2 reviews the scientific literature on neural attention. After a brief discussion of historical works, the development of neural attention for language and vision is discussed. The remainder of the thesis discusses methods developed during my doctoral research. Each method uses some variant of neural attention to solve problems in the fields of document analysis, cancer research and image compression. The fields were chosen to show different aspects of long-range dependency problems. Chapter 3.1 presents a method for detecting text lines in historical documents with handwritten text. The method shows how visual attention can be used to detect linear sequences in images. Chapter 3.2 presents a Transformer model for detecting cancer cells in bone marrow samples from blood cancer patients using global sample information. The data, a type of high-dimensional point cloud, is unsuitable for traditional approaches, demonstrating the versatility of neural attention. In Chapter 3.3, two methods for stereo image compression are presented. The methods show how cross-attention can be used to model redundancies between two images of a stereo-image pair. Finally, a conclusion is given in Chapter 4.

CHAPTER 2

Related Work

In the human brain, attention is a complex neural function that allows perceptual focus on a phenomenon or object while filtering out external stimuli [NZY21]. Biological attention mechanisms can be divided into two types based on what drives attention [TCW+95, NZY21]. In bottom-up unconscious attention, also called saliency attention, external stimuli drive the target of attention. For example, a bright neon light, typically used for emergency exit signs, attracts attention in a dimly lit room such as a cinema. The max-pooling operation can be considered an algorithmic realization of this type of attention. The second type of attention is top-down conscious attention, also called focused attention. Here, attention is task-driven and consciously focused on a specific phenomenon or object. An example would be a person reading a line of text. Methods typically categorized as *neural attention*, such as the self-attention operation in the popular Transformer network [VSP+17a], are of the second type.

The remainder of this chapter is structured as follows: A brief discussion of early work on attention is given in Section 2.1, followed by a discussion of visual attention in Section 2.2 and self-attention in Section 2.3. Section 2.4 provides an overview of Transformers in vision. Finally, recent developments in optimizing the runtime/reducing the complexity of Transformers are discussed in Section 2.5.

2.1 Early Work

Initial works of modelling attention are motivated by early discoveries in neuroscience [DD95] that understood attention as a filter, shaping neural activity to allow for efficient processing of relevant visual information. Inspired by such research and attention in the early primate visual system, Itti et al. [IKN98] propose the concept of a saliency map that selects specific locations in an image based on multiscale image features. The authors apply linear filters to extract colour, intensity and orientation features at 9 different scales. These extracted features are then used to compute a set of feature maps, each capturing a different aspect of the visual scene. These feature maps are then normalized and combined into a single saliency map, highlighting the most selected areas of the image. The saliency map is computed in a parallel and distributed manner, allowing for fast scene analysis.

More recently, in 2014, Bahdanau et al. [BCB15] proposed attentional mechanisms for text data. The authors introduce a new approach to machine translation that allows the model to focus on (or "pay attention to") different parts of the source sentence at different translation stages. This new method contrasts with conventional sequence-to-sequence models that attempt to encode the entire source sentence into a fixed-size vector. The approach works by conditioning the output at each step i on the previous output y_{i-1}, the hidden state s_i, and a context vector c_i resulting from an attention operation on the input sequence.

$$p(y_i|y_1,\dots,y_{i-1},x) = g(y_{i-1},s_i,c_i). \tag{2.1}$$

Here g denotes the RNN. The context vector c_i is the weighted sum of encoded representations of the inputs $h_1,\dots,h_n$:

$$c_i = \sum_j \alpha_{ij} h_j \tag{2.2}$$

with the weights

$$\alpha_{ij} = \frac{exp(a_{ij})}{\sum_k exp(a_{ik})} \tag{2.3}$$

where

$$a_{ij} = a(s_{i-1},h_j) \tag{2.4}$$

computes the similarity between the last hidden state s_{i-1} and the encoded inputs h_j (a_{ij} is in [BCB15] called the *alignment model* and denoted with e_{ik}). The similarity is computed with a neural network trained with the rest of the model. See Figure 2.1 for an overview of the abovementioned mechanism. This architectural design allows access to the complete input sequence in every step. This is in stark contrast to a naive RNN formulation that processes the input sequence sequentially and, therefore, does not have access to future elements and strongly favours locality.

A different approach to model attention in the visual domain is proposed by Mnih et al. [MHGK14]. In their work, Mnih et al. model attention with a sequential decision process of a goal-oriented agent interacting with a visual environment. Instead of processing a whole image at once,ural network f_h is trained to select a sequence of specific regions to focus on, akin to the functioning of the human visual system. At each step, a *glimpse sensor* ρ extracts multiple resolution patches $\rho(\mathbf{x}_t, l_{t-1})$ around the current region $l_{[t-1}$ of focus. These patches are then processed with the *glimpse network* f_g that encodes the extracted patches and current location into the glimpse representation g_t. In every step, the RNN f_h takes the glimpse representation as input together with the previous hidden state h_{t-1} to produce the new internal state h_t. Finally, based on the hidden state, in every step, the *location network* f_l predicts the next location and the *action network* f_a predicts the next action/classification. The full architecture can be seen in

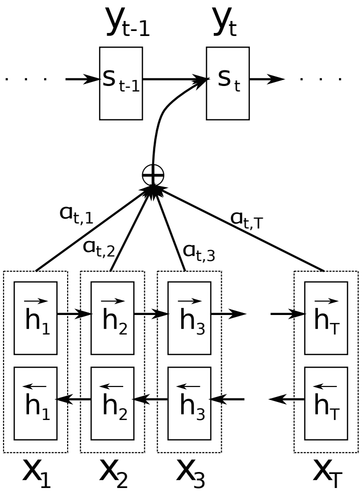


Figure 2.1: Diagram of a single decoding step in attention-based neural machine translation. Taken from Bahdanau et al. [BCB15].

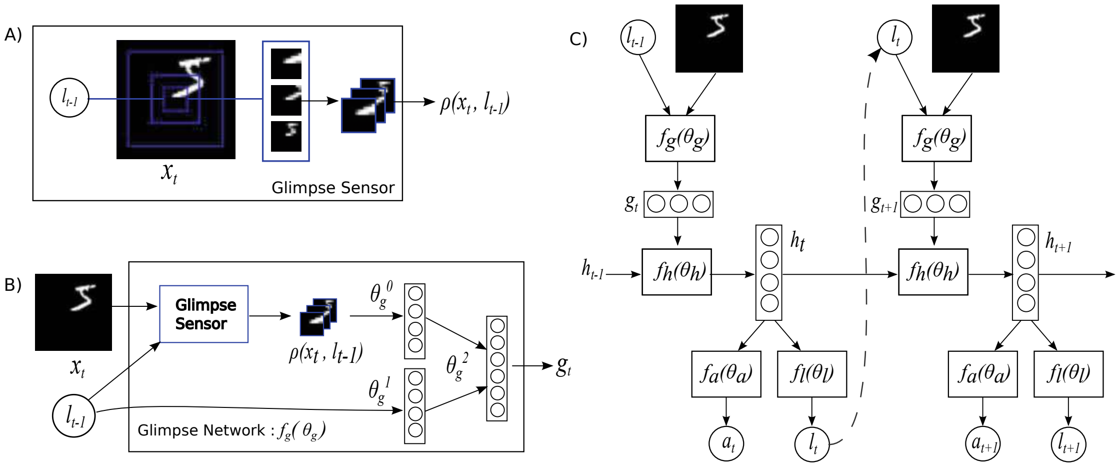
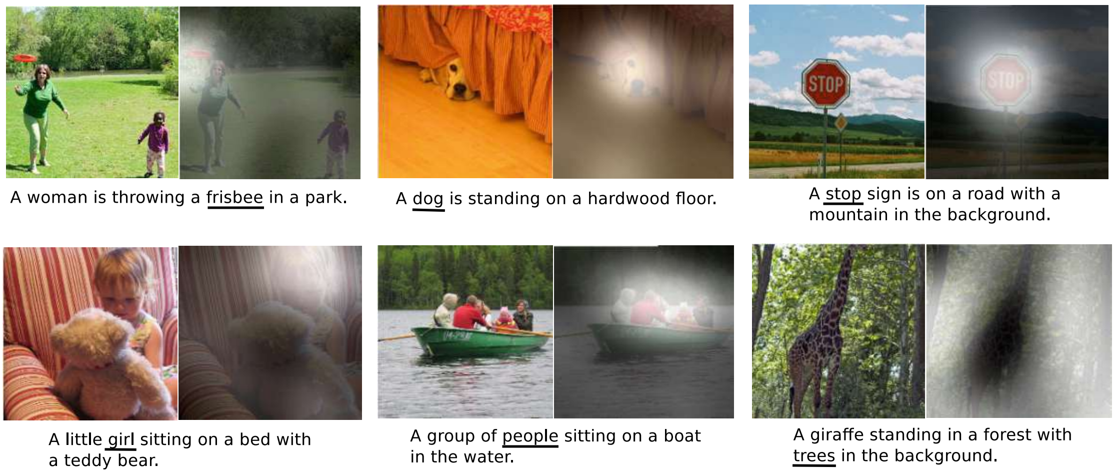


Figure 2.2: An overview of the method proposed in Mnih et al. [MHGK14]. Taken from the original paper.

Figure 2.2. The training process is formulated as a reinforcement learning problem and trained with REINFORCE [Wil92], optimizing attention policies and classification tasks simultaneously. The proposed approach leads to significant computational efficiency, as the model learns to focus on the most informative parts of the image, thereby avoiding the need to process the entire image at high resolution.

The works above build the archetypes for a large part of the literature on neural attention. Bahdanau et al. [BCB15] forms the basis of self-attention and the Transformer [VSP$^+$17a] family of models (see Section 2.3 and Section 2.4) and Mnih et al. [MHGK14] presents the general idea of works on visual attention.



A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Figure 2.3: Examples for semantically correct attention. White regions indicate attention focus. Underlined words show the corresponding word. Taken from Xu et al. [XBK+15].

2.2 Visual Attention

Following the seminal work of Mnih et al. [MHGK14], Lei Ba. et al. [BMK15] extend the idea to real-world image tasks and multiple object recognition in a single image. The authors introduce a model that employs Long-Short-Term Memory units [HS97] to process a sequence of glimpses (local image patches) to localize and recognize multiple objects in an image. The attention mechanism allows the model to dynamically decide which regions of the image to focus on, thereby improving the efficiency and accuracy of object recognition. For training, the model only requires class labels.

Another line of research aims to apply visual attention to image captioning. Xu et al. [XBK+15] propose a method for image captioning that consists of a CNN feature extractor and an LSTM with attention to predicting the caption. In the encoding step, they extract features using the intermediate layer outputs of a VGG [SZ14] network. Due to downsampling in the VGG network, this results in separate image features for each of the 14×14 image patches of the input image. During decoding, the recurrent neural network employs the attention mechanism proposed in Bahdanau et al. [BCB15] to generate a caption while attending to specific locations of the input image. The authors distinguish between *soft* and *hard* attention. In soft attention, the network attends to each location by computing a weighted sum of the features, with the weights describing the importance of a given location for the final prediction. The resulting method is differentiable and can be optimized end-to-end with backpropagation. In hard attention, the network attends to only one region at each step, and REINFORCE is used to train the network. Figure 2.3 shows visualizations of the resulting attention patterns.

Zhu et al. [ZGBFF16] propose a similar method for grounded question answering in images. They encode the image with a CNN and then create a joint encoding of the image and question by processing it with an LSTM. In the decoding stage, the model picks an answer from multiple choices based on its memory while attending to localized VGG image features with a soft attention mechanism. A similar method is introduced by Yang et al. [YHG+16], where they propose Stacked Attention Networks (SANs), which use multiple layers of attention mechanisms to refine query-related image regions iteratively. Chen et al. [CZX+17] additionally use channel-wise attention connections in the CNN.

Finally, some works use visual attention to generate images [Gra13, GDG+15]. Here, a pair of encoding/decoding RNNs is trained as a variational autoencoder to iteratively construct images through an accumulation of modifications. Additionally, the model is able to selectively attend to parts of the scene while ignoring others.

2.3 Transformers

In 2017, Vaswani et al. [VSP+17a] proposed the Transformer architecture, a feed-forward neural network capable of processing an entire sequence in a single forward pass rather than iteratively with an RNN. Figure 2.4 shows an overview of the Transformer architecture.

The model consists of an encoder and a decoder, each consisting of a sequence of 6 identical layers. Each of these layers has two sub-layers. The first is a scaled dot-product self-attention layer, and the second is a position-wise fully connected (i.e. 1D convolutional) network with two layers. Both layers have residual connections and layer normalization [BKH16]. Self-attention can be understood as a special case of attention. For a set of d_k dimensional queries $Q \in \mathbb{R}^{N \times d_k}$, keys $K \in \mathbb{R}^{N \times d_k}$ and d_v dimensional values $V \in \mathbb{R}^{N \times d_v}$ the attention operation

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{2.5}$$

performs a weighted mean of the values V, where the weights are the similarities between the query/key pairs. The softmax is over all keys and ensures that the weights sum to one. The division by $\sqrt{d_k}$ normalizes the attention scores and improves the stability of the gradient flow. Here, the similarity function is the dot-product of queries and keys scaled by the square root of the dimension d, which is why this particular version is called scaled-dot-product-attention. In the case of self-attention all three matrices, Q, K and V are derived from the d_{model} dimensional input $X \in \mathbb{R}^{N \times d_{\text{model}}}$ via linear transformations QW_i^Q, KW_i^K and VW_i^V with the weight matrices $W^Q, W^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$. In addition to that, the self-attention operation in [VSP+17a] is applied multiple times in parallel with different learnable weights, called multihead-attention. From the input, a set of h query, key and value matrices are computed with different weight matrices W_i^Q, W_i^K, W_i^V each. After the self-attention operation, each of these *heads* is combined with a fully connected layer W^O that receives all heads as input (see Figure 2.5).

$$\mathrm{MultiHeadAttn}(Q, K, V) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)W^O \tag{2.6}$$

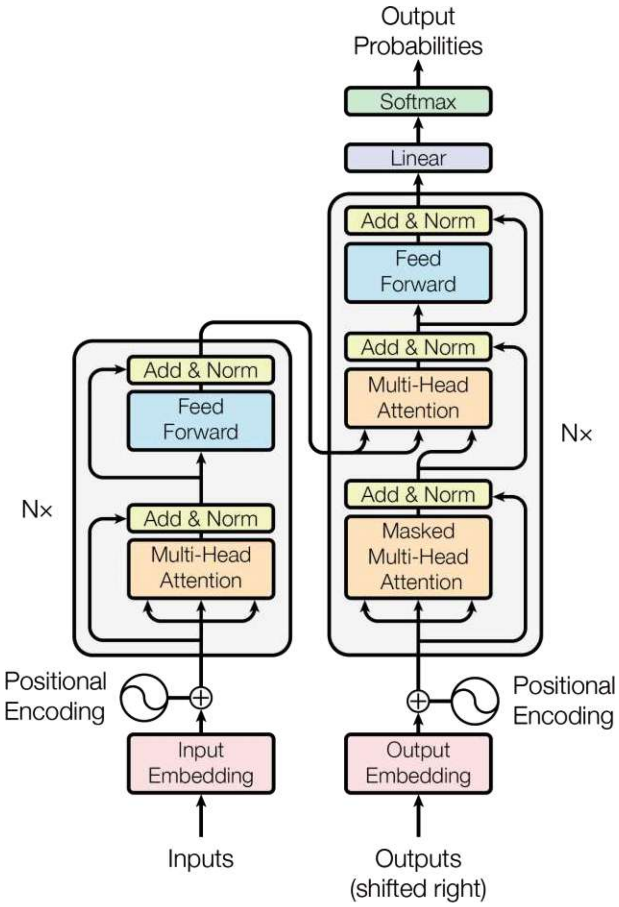


Figure 2.4: An overview of the Transformer architecture. Taken from Vaswani et al. [VSP+17a].

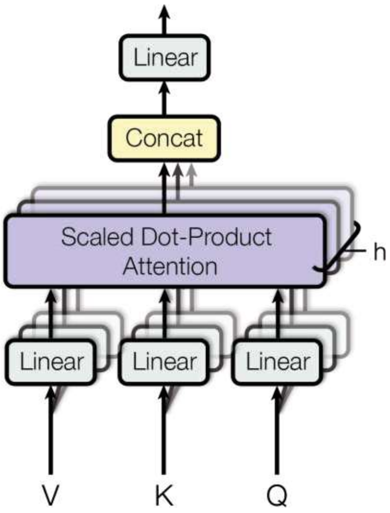


Figure 2.5: The Multihead attention operation. Taken from Vaswani et al. [VSP+17a].

where

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V). \tag{2.7}$$

The complete scaled-multi-head-dot-self-attention is then

$$\text{MultiHeadSelfAttention}(X) = \text{MultiHeadAttn}(X, X, X). \tag{2.8}$$

The Transformer encoder consists solely of self-attention layers. The decoder also contains cross-attention connections where the keys and values come from the output of the encoder, and the queries come from the output of the previous decoder layer. After each attention layer, a two-layer position-wise fully connected network is applied (see Figure 2.4). The Transformer model was originally proposed for machine translation, i.e. a sequence-to-sequence task, which requires two additional modifications to the architecture described above. During training, the attention operation of the decoder is prevented from cheating by attending to future inputs. This is done by masking non-causal query/key combinations in the attention matrix

$$A = \operatorname{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right). \tag{2.9}$$

Furthermore, the Transformer architecture, as described above, is fully permutation equivariant, meaning any encoding of information in the order of inputs in the sequence cannot be modelled with this architecture alone. To solve this issue, the input is augmented with a positional embedding before being fed into the first layer. There are different variants of positional encoding, both learned and fixed [DSS22]. In the original formulation of the Transformer [VSP+17a], a fixed Fourier embedding PE is added to each input X

$$\mathrm{PE}(n, i) = \begin{cases} \sin\left(\frac{n}{10000^{2i/d_{\text{model}}}}\right) & i \text{ even} \\ \cos\left(\frac{n}{10000^{2i/d_{\text{model}}}}\right) & i \text{ odd} \end{cases} \tag{2.10}$$

where n denotes the position in the sequence, i the dimension and d_{model} the dimensionality of the input X.

The Transformer model is, in principle, capable of modelling arbitrary long sequences. However, in practice, the required memory is the bottleneck when scaling to longer input sequences. The attention mechanism requires pair-wise similarity scores for queries and keys, which leads to quadratic $\mathcal{O}(n^2)$ memory and runtime complexity. Possible solutions for these issues are discussed in Section 2.5. The original Transformer model was proposed for machine translation [VSP+17a] and contained both encoder and decoder. It is trained in a supervised manner on English-German sentence pairs. Follow-up work explored options for general pre-training of Transformers.

The BERT model [DCLT19] is an encoder-only Transformer pretrained on a large corpus of text data. During training, the model receives text where some input positions are masked. The model is then supposed to predict the correct token for every masked position. BERT can be used for classification tasks by processing the output of a separate *[cls]* token in a fully connected network that is fine-tuned for specific tasks. For the time, the proposed BERT model was considered very large, with 340 million parameters. DistilBERT [SDCW19] is a distilled BERT model optimized for efficient deployment. The model has fewer parameters than BERT and only a small drop in performance. ALBERT [LCG+19] is another BERT-like model with a reduced parameter count that uses parameter sharing across layers. Other approaches modify the training process. RoBERTA [LOG+19] essentially scales up the training and achieves improved performance over BERT. ELECTRA [CLLM20] aims to improve efficiency by reformulating the pre-training objective to a discriminative task. Instead of predicting masked tokens, they employ a small generator network that perturbs the input and trains the ELECTRA model on the discriminative task of detecting corrupted tokens. The BERT model also forms the basis of the Vision Transformer [DBK+20] that is discussed in Section 2.4.

An alternative to the masked bidirectional pre-training proposed in the BERT paper [DCLT19] is pre-training with autoregressive next-token prediction. The GPT models [RWC+19, BMR+20a, Ope23] are decoder-only Transformers trained on a large corpus of text in an autoregressive manner, i.e. given a sequence of tokens, predict the next token. They observe consistent improvements when scaling both model size and dataset size. The trained models show emergent behaviour and zero-shot capabilities.

2.4 Transformers for Vision

While initially only proposed for natural language processing, recent works apply networks containing self-attention operations to image data. Wang et al. [WGGH18] propose non-local neural network modules that can be understood as a generalization of the self-attention operation. The non-local operation is defined as

$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_j f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j) \tag{2.11}$$

for input $(\mathbf{x}_i)_{i \in I}$ and output $\mathbf{y}_i$ at position i (in space for images or spacetime for videos). f and g are typically learned during training. $\mathcal{C}(\mathbf{x})$ is a normalization factor. for

$$g(\mathbf{x}) = W_g \mathbf{x} f(\mathbf{x}_i, \mathbf{x}_j) = e^{(W_\theta \mathbf{x}_i)^\top W_\phi \mathbf{x}_j} \tag{2.12}$$

the operation is identical to the self-attention operation from Vaswani et al. [VSP+17b]. The resulting module is capable of modelling long-range dependencies but suffers from the quadratic complexity of the attention operation. To circumvent this issue, Ramachandran et al. [RPV+19] employ local self-attention. They replace every convolution in a ResNet [HZRS16] with a self-attention operation restricted to a local rectangular window to mimic the local receptive field of a convolution. The model performs best if the initial convolutions in the ResNet, the *stem*, are kept convolutional while the remaining convolutions are replaced with local self-attention operations. Unlike these approaches that add self-attentional components into CNNs or CNN-inspired architectures, the Vision Transformer (ViT) [DBK+20] directly applies the Transformer model from [VSP+17b] to images. Inspired by the BERT model [DCLT19], the ViT consists of a Transformer encoder with an additional class *[cls]* token applied to a sequence of image patches. The model architecture can be seen in Figure 2.6. To process a 2D image with a Transformer, which was initially introduced for sequence modelling[1], the image is split into patches, and each patch is flattened and processed with a linear layer. The resulting patch encodings are used as input for the BERT-like encoder-only Transformer. Learnable 1D positional encodings are used to embed the positional information. The model was initially proposed for image classification. The ViT shows only modest results when compared to CNNs when trained on mid-sized datasets like ImageNet [DDS+09, HWC+22] but outperforms them when pre-trained on larger datasets like JFT-300M. For smaller

[1] The order information is encoded via positional encoding.

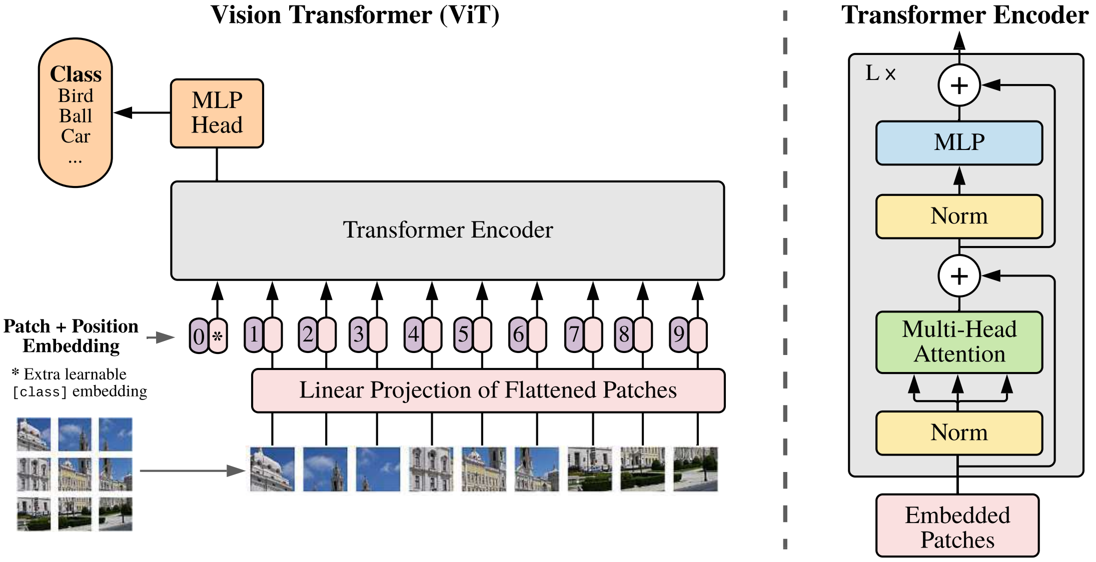


Figure 2.6: The ViT architecture. Taken from [DBK+20].

datasets, the inductive locality bias of convolutions is beneficial when pre-trained on a sufficient scale. Transformers are capable of learning higher quality intermediate representations [RUK+21, HWC+22]. Contrary to CNNs, the ViT is capable of modelling long-range dependencies in a single layer. However, since the initial patches are projected to embedding vectors with a simple linear layer, the architecture can struggle with smaller local structures. Multiple follow-up works tackle this issue. TNT [HXW+21] further divides the patches into sub-patches and introduces a Transformer-in-Transformer architecture to utilize these sub-patches. SwinTransformer [LLC+21, LHL+22] performs local attention in a window and applies different window shifts for different layers to allow for cross-window connections. Others apply the vision Transformer architecture to settings other than image classification. The Detection Transformer DETR proposed by Carion et al. [CMS+20] treats the object detection task as a set prediction problem. DETR encodes an image with a CNN followed by a Transformer encoder. In the decoder, the network performs cross-attention between a set of learned embeddings, *object queries*. The output of this cross-attention operation is processed with another Transformer and followed by a linear layer that predicts the output. The resulting models perform on par with Faster R-CNN on COCO [LMB+14] but show poor performance on smaller objects [HWC+22]. To address this, Zhu et al. [ZSL+20] proposed Deformable DETR, a variant of DETR. Instead of global cross-attention connections between the object queries and the encoded image, Deformable DETR introduces a *deformable attention module* that only attends to a small set of key positions. The resulting method achieves better performance and faster training and inference speeds. Other works applied ideas from ViT to segmentation tasks [ZLZ+21, WZA+21, WXW+21].

While the original ViT is trained supervised, Radford et al. [RKH+21] propose CLIP, a multimodal model consisting of one Transformer encoding text and a ViT encoding an image trained in parallel with a contrastive loss on image-text pairs. During training, a batch of images is processed with a Vit, and the corresponding batch of captions is processed with the language Transformer. The resulting embedding vectors are compared with dot-product similarity. The training loss ensures that similarities between correct image/caption pairs are high and incorrect ones are low. The resulting model can then be used for zero-shot tasks by probing it with a language prompt. For example, to measure zero-shot accuracy on ImageNet, one can probe the model with *A photo of a {object}*, where *{object}* is replaced by each ImageNet class. Then, the class with the highest similarity between image encoding and prompt encoding can be considered the prediction. CLIP achieves 76.2% top-1 accuracy on ImageNet-1K in a zero-shot setting, i.e. without using any ImageNet training labels.

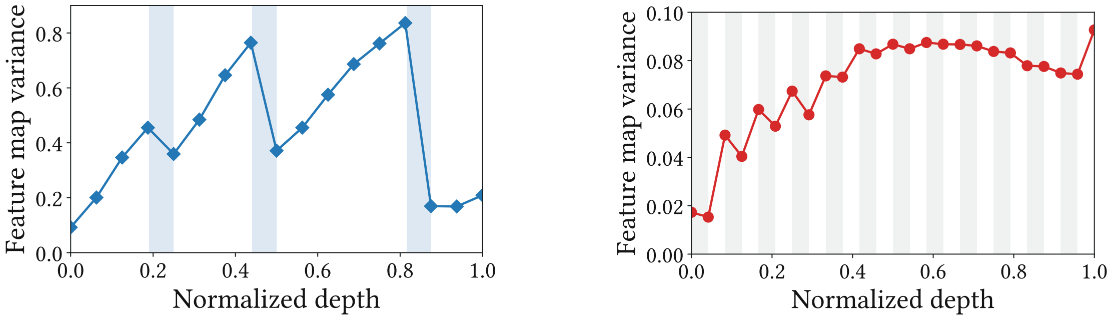
How do Vision Transformers differ from CNNs? CNNs have, by design, an implicit locality prior. This missing prior in ViTs is often assumed to be responsible for the worse performance of vanilla ViTs in general computer vision tasks like object detection or semantic segmentation [LMW+22]. Variants like the SwinTransformer [LLC+21, LHL+22] reintroduce such a local prior to circumvent these issues. When directly comparing Multihead Self-Attention (MSA) with convolutions, it can be shown that these exhibit opposite behaviour in some aspects, as evidenced by Park et al. [PK22]:

Flatter loss landscape MSAs lead to a flatter loss landscape, as measured by the average magnitude of the Hessian eigenvalues during training. A flatter loss landscape is generally associated with improved generalization and robustness [LXT+18], suggesting that ViTs learn better representations than CNNs.

Non-convex loss: The loss function of ViT is non-convex, whereas, for ResNets, it is nearly convex [PK22]. The non-convexity can lead to suboptimal training, particularly during early stages [JSF+20] and for smaller training datasets. This explains the sub-par results of ViTs in these scenarios. This phenomenon explains the subpar performance of ViTs on medium-sized datasets like ImageNet compared to CNNs. Several strategies can mitigate this issue:

- Employing loss landscape smoothing methods
- Utilizing Global Average Pooling
- Restricting MSA to local windows (local attention)
- Training on larger datasets

MSAs act as low-pass filters : Convolutions act as high-pass filters, while MSAs function as low-pass filters. See Figure 2.7 for visualization of the feature variance after different operations. MSA reduces variance, effectively acting as a low-pass filter, while convolutions increase feature variance. This behaviour can be attributed to the aggregation of feature maps in MSA, essentially forming an ensemble of input features. As shown in eq. (2.5), the attention operation computes a weighted sum of the values, which are themselves the image of a linear map applied to the input. Since the softmax caps these weights by 1, preventing the amplification of outlier features and instead mixing them with the remaining features, thus reducing variance. This perspective of MSAs as low-pass filters and convolutions as high-pass filters suggests that a complementary approach combining both architectures may yield superior results in various computer vision tasks [PK22].



(a) ResNet

(b) ViT

Figure 2.7: Convolutions increase feature variance, and MSA reduces variance. White, grey, and blue areas are convolutions, MSA, and subsampling, respectively. Taken from Park et al. [PK22].

2.5 Efficient Transformers

The feed-forward approach of the Transformer is more efficient than sequential processing with an RNN as long as the sequence length is shorter than the dimensionality of the embedding [VSP+17a]. Furthermore, the attention matrix from eq. (1.1) grows quadratically $\mathcal{O}(n^2)$ in size with increasing sequence length n, making available memory a bottleneck for longer inputs, and the computational demand grows quadratically. These issues have been the source of several recent works (for an extended discussion and benchmarks of the different methods, see [TDBM20, TDA+20]). In the Reformer [KKL19] paper, the authors observe that the typical attention matrix is sparse, with only a few positions contributing substantially to the result. They reduce the complexity to $\mathcal{O}(n \log n)$ through locality-sensitive hashing between layers to cluster the inputs and only apply attention locally. Additionally, they work with invertible layers to further reduce memory requirements. Another approach that uses the sparsity of the attention matrix is the BigBird model [ZGD+20] that replaces the full attention operation with a sum of local attention, randomly sampled attention and attention to fixed predefined positions, which results in linearly scaling complexity $\mathcal{O}(n)$. Another line of research works by interpreting the self-attention mechanism as a kernel method [TBY+19] and linearising the kernel k [CLD+20, KVPF20] as shown in eq. (2.13).

$$\begin{aligned}\mathrm{Attn}(Q_i, K, V) &= \mathrm{softmax}(\frac{Q_i K^\intercal}{\sqrt{d}})V \\ &= \frac{\sum_j k(Q_i, K_j)V_j}{\sum_j k(Q_i, K_j)} = \frac{\sum_j \phi(Q_i)^\intercal \phi(K_j)V_j}{\sum_j \phi(Q_i)^\intercal \phi(K_j)}\end{aligned} \tag{2.13}$$

The function ϕ in eq. (2.13) maps to an infinite-dimensional vector space. However, Katharopoulus et al. [KVPF20] show that by instead approximating it with the computationally simpler function $\phi : x \mapsto \mathrm{elu}(x) + 1$, the complexity of the self-attention operation becomes linear $\mathcal{O}(n)$ with only a small reduction in performance. Rather than modifying the similarity function, in Choromanski et al. [CLD+20], the softmax function is approximated with orthogonal random features, which also results in linear complexity $\mathcal{O}(n)$. The Set Transformer [LLK+19b] replaces the self-attention operation with two cross-attention operations to achieve $\mathcal{O}(nk)$ for a fixed $k \ll n$. This is achieved by replacing self-attention blocks with *Induced Set Attention Blocks* (ISAB):

$$\mathrm{ISAB}(X) = \mathrm{MAB}(X, \mathrm{MAB}(I_m, X)) \tag{2.14}$$

where

$$\mathrm{MAB}(X, Y) = \mathrm{LayerNorm}(H(X, Y) + rFF(H(X, Y))) \tag{2.15}$$

$$H(X, Y) = \mathrm{LayerNorm}(X + \mathrm{MultiHeadAttn}(X, Y, Y)) \tag{2.16}$$

with a feed forward network $F(\cdot)$ and $I_m \in \mathbb{R}^{m \times d}$ are m trainable *induced points*. To see why this reduces the complexity from quadratic to linear in the input length, notice that in each application of MAB in eq. (2.14), one of the two inputs has sequence length m making the total complexity $\mathcal{O}(nk + kn) = \mathcal{O}(nk)$. A similar idea is used in the Perceiver model from Jaegle et al. [JGB+21]. The Perceiver uses an asymmetric attention mechanism that iteratively refines inputs into a compressed latent bottleneck, enabling scalability for handling large-sized inputs. The input is connected to the latent stream via multiple cross-attention connections, allowing for efficient processing of large inputs and making it, in principle, modality agnostic. A completely different approach to improve the efficiency of the Transformer is presented by FlashAttention [DFE+22], which introduces an I/O-aware exact attention algorithm designed to enhance the performance of Transformers, particularly when dealing with long sequences. FlashAttention performs tiling to minimize the number of memory operations between the GPU's high bandwidth memory (HBM) and on-chip SRAM. FlashAttention-2 [Dao23] further tweaks the algorithm to reduce the number of non-matrix multiplication floating-point operations (FLOPs), parallelizing the computation of attention across thread blocks (even for a single head) to increase occupancy and distributing the work more effectively within each thread block.

The methods above only give a brief overview of more efficient alternatives to the original Transformer model. Tay et al. [TDBM20] outline a general taxonomy of different approaches shown in Figure 2.8.

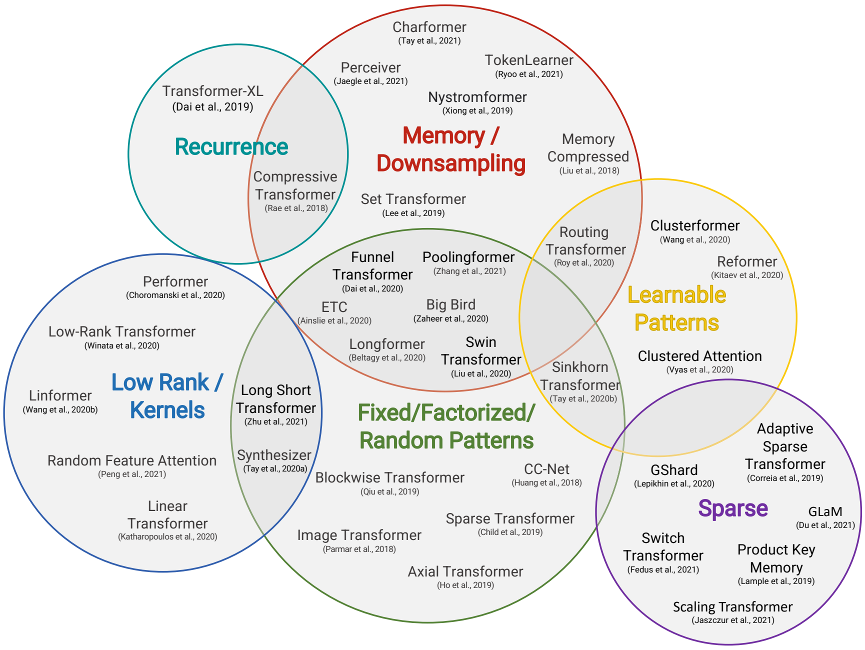


Figure 2.8: Taxonomy of efficient Transformer architectures [TDBM20].

2.6 Summary

This chapter started with a historical overview of neural attention research. Early work aimed to replicate biological attention [DD95, TCW$^+$95] and focused on attention in vision, the work that forms the basis for most modern methods came from NLP [BCB15]. These ideas from Bahdanau et al. [BCB15] were subsequently extended to vision in various visual attention methods [XBK$^+$15, SZ14, ZGBFF16]. In 2017, Vaswani et al. [VSP$^+$17b] proposed the Transformer model, which has since been established as the template architecture for neural attention models. Follow-up work has focused on pre-training strategies [DCLT19] and extending the architecture to vision [DBK$^+$20, LLC$^+$21]. A major bottleneck of the original Transformer is its quadratic complexity, which motivated follow-up work aimed at modifying the architecture to achieve sub-quadratic and even linear complexity [XLC$^+$20, TDBM20], albeit with trade-offs in performance.

CHAPTER 3

Methodology

This chapter discusses the methodology and results of the research conducted for this thesis. Each section focuses on one specific application and begins with a brief discussion of the application's relevant related work, followed by a discussion of the proposed methods, results and a conclusion.

3.1 Modelling Text Baselines with Visual Attention

Handwritten Text Recognition (HTR) is the extraction of machine-readable text from images of handwritten documents. While modern methods allow text extraction with only minor errors for images of modern printed documents allow [HCH$^+$19], historical handwritten documents still present a challenge to current methods [SRT$^+$17]. Recent advances [TW19, MKW17, CMV$^+$19] have seen methods that approach handwritten text recognition (HTR) on entire document pages, bypassing the need for preliminary extraction of text baselines. Nevertheless, the dominant approach to HTR involves the initial extraction of text lines, followed by the conversion of images into textual data via sequential processing of the text lines in a second step[GLS$^+$19]. Errors in text baseline segmentation often cause HTR inaccuracies, highlighting the need for robust baseline extraction techniques [SRT$^+$17].

This thesis introduces a novel method that uses a recurrent CNN to determine baseline coordinates from document images, using only start points and angles as inputs. The proposed model employs visual attention to follow the linear structure of textual baselines in document images. Instead of processing the entire image, the method uses a small recurrent subnetwork that focuses on specific locations in the image and processes baseline sequences linearly. This methodology represents an alternative to traditional approaches to text baseline extraction as it eliminates the need for heuristic post-processing steps. It includes a technique to determine start points and angles through a semantic segmentation CNN, which then assists the recurrent CNN in extracting baseline coordinates. Both components allow end-to-end training by using only baseline coordinates as labels. Contrary to heuristic post-processing steps used in other methods [GLS+19], this allows for rapid adaptation to new document layouts and types by incorporating additional training examples without requiring modifications of the underlying algorithms.

Contrary to the remainder of this thesis, the model described in this section is not based on self-attention or a related mechanism. Instead, the proposed method can be more broadly categorized as falling in the visual attention category (as described in Section 2.2), where an input image is processed in a sequence or local windows instead of all at once, similar to Xu et al. [XBK+15]. Such a linear sequence of local attention (*glimpses*) is a natural fit for text baselines, which are themselves linear structures in possible larger images.

The approach focuses on historical documents since these provide the most significant challenges [DKF+17, DKSG19]. The model is trained and validated on the cBAD 2019 dataset [DKSG19]. An example prediction on an image from the cBAD 2019 test set demonstrates the predictive capability of the system (see Figure 3.1).

The following sections describe other baseline recognition efforts and explain the proposed methodology and results on the cBAD dataset.

Note: This Section is based on work previously published in the Proceedings of the International Conference for Pattern Recognition (ICPR) 2020[WS21]. The text has been adapted and, in some cases, except stated otherwise, directly reproduced to maintain the precision and specificity of the original work. All figures are taken without modification from the cited paper.

3.1.1 Related Work

Text-baseline recognition of historical documents has seen significant contributions in recent years, particularly with the introduction of CNNs. Methods using Fully Convolutional Networks (FCN) [LSD15] have proven effective by treating baseline recognition as a pixel-wise classification challenge, especially on historical datasets [GLS+19, BDKES18, FLMS18, SHM+18]. In these CNN-based approaches, the segmentation process is followed by heuristic post-processing methods to derive baseline coordinates from the segmentation masks. For example, Gruning et al. [GLS+19] present a two-stage approach using an adapted U-Net with residual connections and an attention mechanism, followed by a bottom-up clustering approach to extract the baseline coordinates. Schone et al. [SHM+18] use an FCN to predict the entire perimeters of text lines. Parallel developments in HTR demonstrate methods that infer text baselines as an integral part of HTR, such as Wigington et al. [WTD+18], who implement a three-network system that determines start points, predicts text line coverage, and extracts text in a single end-to-end optimizable pipeline.

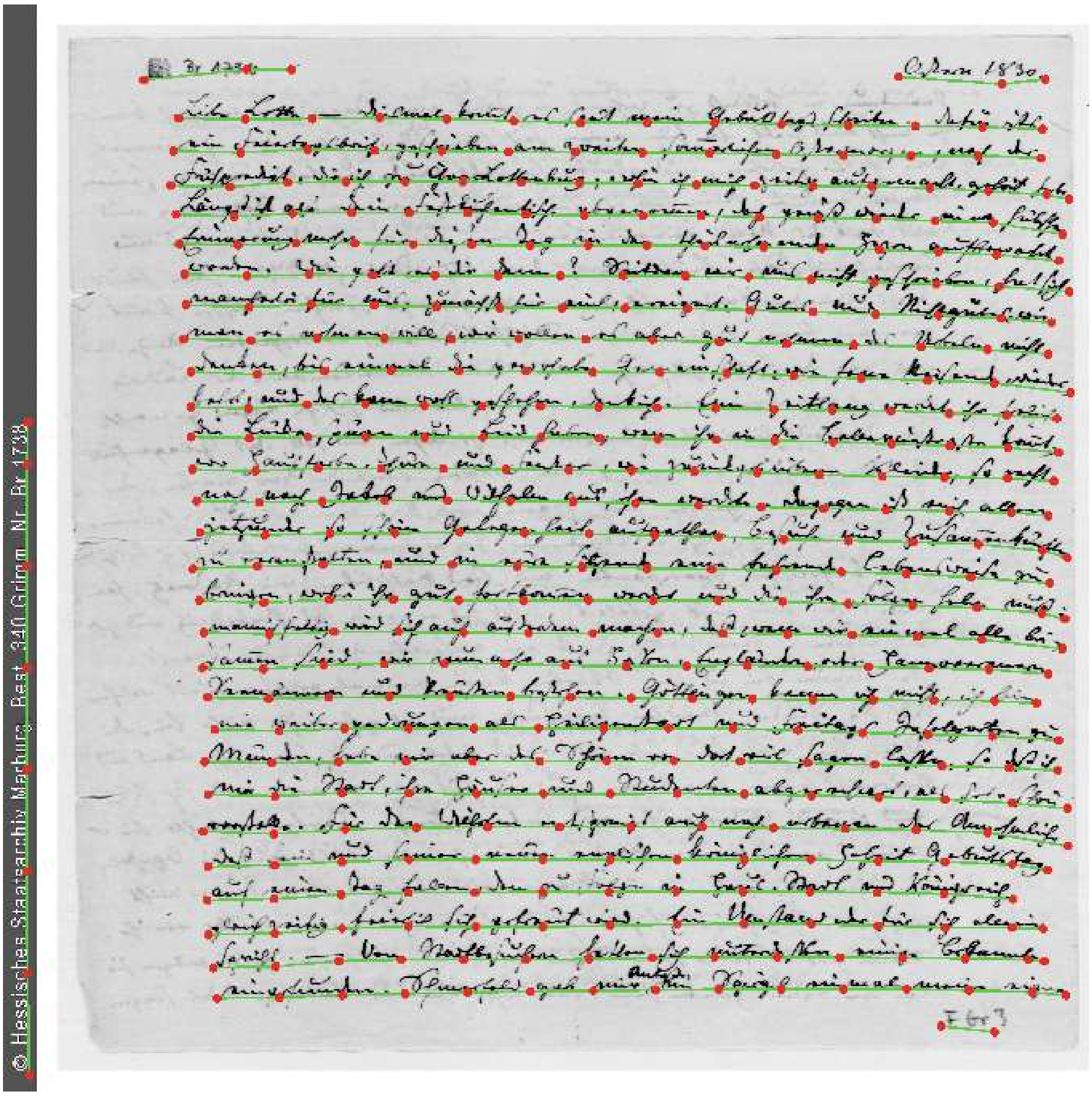

© Hessisches Staatsarchiv Marburg, Best. 340 Grimm, Nr. Br 1738

Figure 3.1: A sample output of the proposed method on a document from the cBAD test set. The red dots denote the predicted baseline coordinates, and the green lines depict the resulting baseline.

3.1.2 Methodology

Extracting baseline coordinates from document images is an essential step in document image analysis. The proposed method applies two steps for effectively extracting text baselines as coordinates:

1. Segmentation of the input image: This involves pixel-wise analysis to identify starting points, termination points, baseline regions, and text regions.
2. Retrieving the coordinates: Commencing from identified starting points, the method follows the baselines to capture the coordinate data.

This procedure is designed to be modular, allowing for the substitution of the initial point and angle generation with alternative techniques.

3.1.2.1 Segmentation of Baselines

The recurrent CNN presented in Section 3.1.2.2 relies on baseline starting points for extraction. To obtain baseline start point candidates (in the typical case where they are not provided), a semantic segmentation model is applied to the page image, yielding pixel-wise predictions for the following classes *start-point, endpoint, baseline, text, border,* and *background.* Including the *text* class, which includes a region of predetermined height above the baselines, improves model learning and avoids segmentation gaps. Meanwhile, the *border* class, defined as a filled sphere of uniform width around the start and endpoints, helps to distinguish proximal points during training but is excluded post-training.

The CNN used for segmentation is based on the U-Net [RFB15] and Large Kernel Matters architectures [PZY$^+$17] with a ResNeXt50 encoder [XGD$^+$17] pre-trained on ImageNet. ResNeXt50 was chosen because it provides a reasonable trade-off between performance and size compared to other methods on the cBAD 2019 dataset. The upsampling operation is performed using nearest neighbour upsampling, followed by 3×3 convolutions instead of deconvolutions to avoid checkerboard artefacts [ODO16]. The network is augmented by a Global Convolutional Network (GCN) module and a boundary refinement module [PZY$^+$17], where the GCN consists of parallel $1 \times k$ and $k \times 1$ convolutions, with $k = 9$ and $c = 25$. Additionally, to aid the recurrent CNN (see Section 3.1.2.2) during the later recurrent prediction stage, the segmentation maps are combined with the input image to form a three-channel image consisting of the greyscale document image and the segmentation maps for baseline and text (see Figure 3.2).

3.1.2.2 Retrieval of Baseline Coordinates

The next stage involves the *Line Rider* model, a dual recurrent CNN framework designed to predict the coordinates of a baseline and identify its endpoint. At each step, the model processes an image patch with two CNNs.

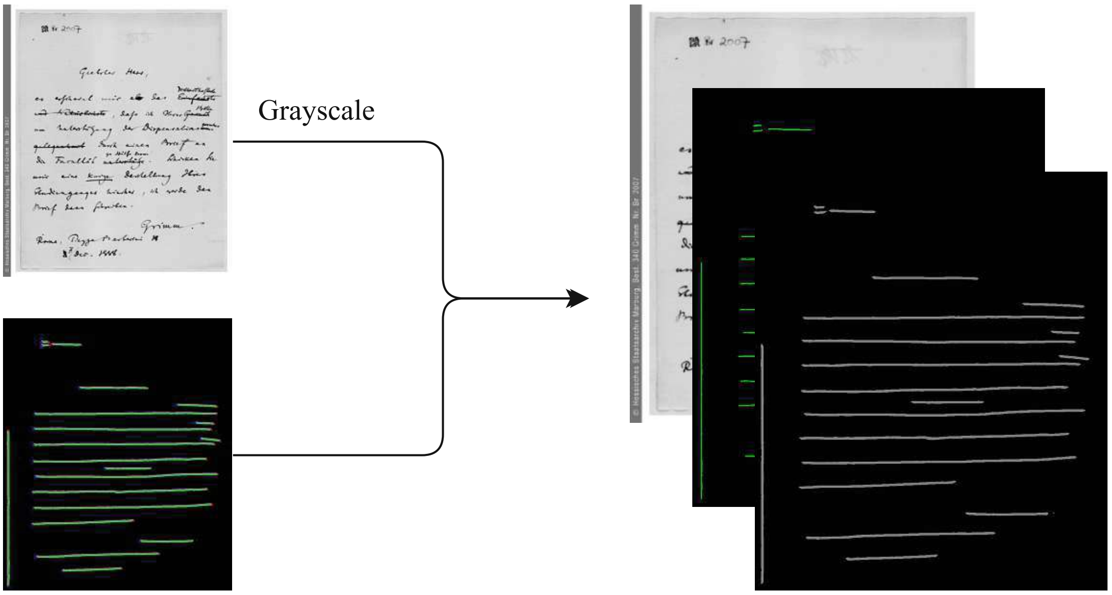


Figure 3.2: The segmentation output for baselines and text is combined with the grayscale input image to form a three-channel tensor.

For each baseline, the model is initialized with the starting point, the starting angle and the estimated window size, which corresponds to the scale of the text and the distance to adjacent baselines. Image patches are extracted and transformed into a 32×32 pixel input to the CNNs. The angle predictor CNN uses a regression approach to predict the sine and cosine components of the angle, resulting in more accurate results than direct angle prediction.

The extracted image patch size and the length of each baseline segment are determined by the window size. This is calculated individually for every baseline and should reflect the writing size and the spacing between adjacent baselines. As the cBAD dataset lacks annotations for text heights, we estimate the window size using the distance between the start point of the baseline and the start points of adjacent lines. Using these parameters, we extract an image region with a height equal to the window size, a width four times the window height, and the appropriate angle, positioning the start point at the center of the left window border.

Based on the start points (x_n, y_n), window angle α_n and the predicted values $\sin_{n+1}$ and $\cos_{n+1}$ and using the trigonometric identities $\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$ and $\sin(a+b) = \sin(a)\cos(b) + \cos(a)\sin(b)$ the next baseline coordinate (x_{n+1}, y_{n+1})

Layer	Filter size	Padding	Channels
Input	/	/	3
Conv2d	3×3	0	16
Conv2d	3×3	2	32
MaxPool2d	2×2	0	32
Conv2d	3×3	0	64
Conv2d	3×3	2	64
MaxPool2d	2×2	0	64
Conv2d	3×3	0	128
Conv2d	3×3	0	128
Conv2d	3×3	0	128
MaxPool2d	2×2	0	128
Flatten	/	/	128
Linear	/	/	2

Table 3.1: The CNN that is used to predict the angle of the next window. Next Window Predictor CNN in Figure 3.3. ReLUs are used after every convolutional layer, and tanh is used in the output layer.

and window angle α_{n+1} is computed according to

$$\begin{aligned} x_{n+1} &= x_n + s \cdot [\cos(\alpha_n)\cos_{n+1} - \sin(\alpha_n)\sin_{n+1}], \\ y_{n+1} &= y_n - s \cdot [\sin(\alpha_n)\cos_{n+1} + \cos(\alpha_n)\sin_{n+1}], \\ \alpha_{n+1} &= \alpha_n + \arctan\left(\frac{\sin_{n+1}}{\cos_{n+1}}\right), \end{aligned} \tag{3.1}$$

where s denotes the step size. s is set to half the width (and therefore also two times the height of the input window). A step size larger than half the width can exceed the baseline end if it is not in the current window. Furthermore, a step size that is too large increases prediction error in case of incorrectly predicted angles. A small step size, on the other hand, makes the method more computationally demanding. The chosen step size of half the width was empirically found to provide the best trade-off. An adaptive step size could provide further performance improvements, but in initial experiments, it made the model less stable since the model collapsed to very small baselines in some cases.

During training, the ground truth baseline is split into segments of constant length *step size*. The entire process is differentiable. The model architecture is described in Table 3.1.

The baseline endpoint is determined by a separate CNN, described in Table 3.2, which is optimized for determining whether the current baseline terminates at a given position. The architecture of this network is designed to predict the location of the endpoint and the length of the last segment. The length prediction is a regression task with a mean squared error loss, and the endpoint detection is trained as a binary prediction problem with a binary cross-entropy loss. The ground truth labels for endpoint detection are

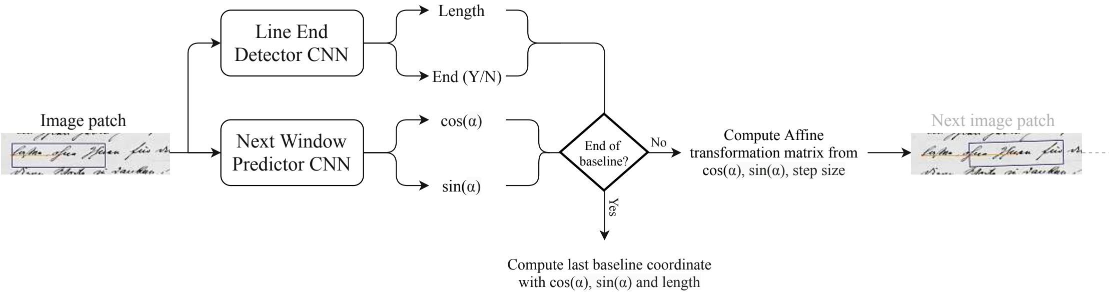


Image patch
Line End Detector CNN
Next Window Predictor CNN
Length
End (Y/N)
cos(α)
sin(α)
End of baseline?
No
Yes
Compute Affine transformation matrix from cos(α), sin(α), step size
Next image patch
Compute last baseline coordinate with cos(α), sin(α) and length

Figure 3.3: One step in the Line Rider Network. The input image patch is fed into the two CNNs. The line-end-CNN 3.2 predicts the end of the baseline and the length of the last baseline segment, while the next-window-CNN 3.1 predicts the angle of the next window. If the end of the baseline is reached, the last baseline coordinate is computed from the predicted values. Otherwise, the length output of the line-end-CNN is ignored, and the next window is extracted according to the output of the next-window-CNN.

Layer	Filter size	Padding	Stride	Channels
Input	/	/		3
Conv2d	3×3	0×1	/	32
Conv2d	3×3	0×1	1×1	64
MaxPool2d	3×3	0	/	64
Conv2d	3×3	0×1	1×1	128
Conv2d	3×3	0×1	1×1	128
Conv2d	5×1	0×1	/	128
Flatten	/	/		$10 \cdot 128$
Linear	/	/		2

Table 3.2: The CNN used to predict the end of baselines and the length of the last segment. Line End Detector CNN in Figure 3.3. ReLUs are used after every convolutional layer and sigmoid function on the output layer.

smoothed with a Gaussian kernel to account for baselines where the end is ambiguous or close to segment boundaries.

The iterative mechanism of window extraction and subsequent prediction is graphically represented in Figure 3.3.

3.1.2.3 Training

The proposed model was trained and evaluated on the cBAD 2019 dataset [DKSG19], which contains 3028 images from historical documents with diverse layouts and writings. These images were pre-processed and labelled with text regions, baselines and specific image areas. The dataset was divided into training, evaluation, and test subsets containing

755, 755, and 1511 images respectively. Text-baseline annotations were represented as polygonal chains.

Segmentation Prior to segmentation, the images were pre-processed by rescaling to standardize the minimum side to a length of 1024 pixels. The model was trained for 80 epochs using a cross-entropy loss function and the Adam optimizer [KB14] with a learning rate of $4 \cdot 10^{-4}$. During training, the images were randomly cropped to 1024 × 1024 pixels. Random rotations were employed as data augmentation with a rotation within $\alpha \in [-20°, 20°]$ and a 70% in 30% of cases and $\alpha \in \{-90°, 0°, 90°\}$ otherwise.

Labels for baseline training were created by dividing baselines into segments of constant width and creating polygonal regions of fixed height for text annotation. The start and endpoints were labelled with solid circles and outlined by rings surrounding these points.

Line Rider Model The Line Rider architecture was configured to process full-page images, each rescaled to 1024 × 1024 pixels.

Two different training scenarios were evaluated: 1) direct use of the original raw document image and 2) enhancement of the document image with the baseline segmentation output. In the latter case, the segmentation results were merged with the document image by stacking the probability masks for both *baseline* and *text* classifications in the channel dimension with the greyscale image representation, as shown in Figure 3.2.

During the initial training phases, the model tends to diverge from the correct baseline when making incorrect predictions. This happened in particular in the early stages of training. To mitigate the effect of this on longer baselines, the coordinates in the recurrent process were reset from the predicted to the ground coordinates each n^{th} iteration, with n incremented by one after every 400 steps, capped at 8.

Data augmentation was applied, consisting of random perturbations to the window dimensions, starting points, angles, and predicted coordinates for each image fragment.

The model was trained with a learning rate of $4 \cdot 10^{-4}$ for only 5 epochs. The model converges very quickly due to its small size (see Table 3.1 and Table 3.2).

3.1.3 Evaluation

The performance of the proposed Line Rider model is evaluated on the cBAD 2019 test set[DKSG19] in Section 3.1.3.1 and the impact of additional segmentation information in Section 3.1.3.2.

3.1.3.1 Model Configurations and Performance

For the quantitative evaluation of the predictions, the evaluation tool provided as part of the cBAD 2019 competition was used[1]. This script checks whether the predicted baselines

[1] https://github.com/Transkribus/TranskribusBaseLineEvaluationScheme

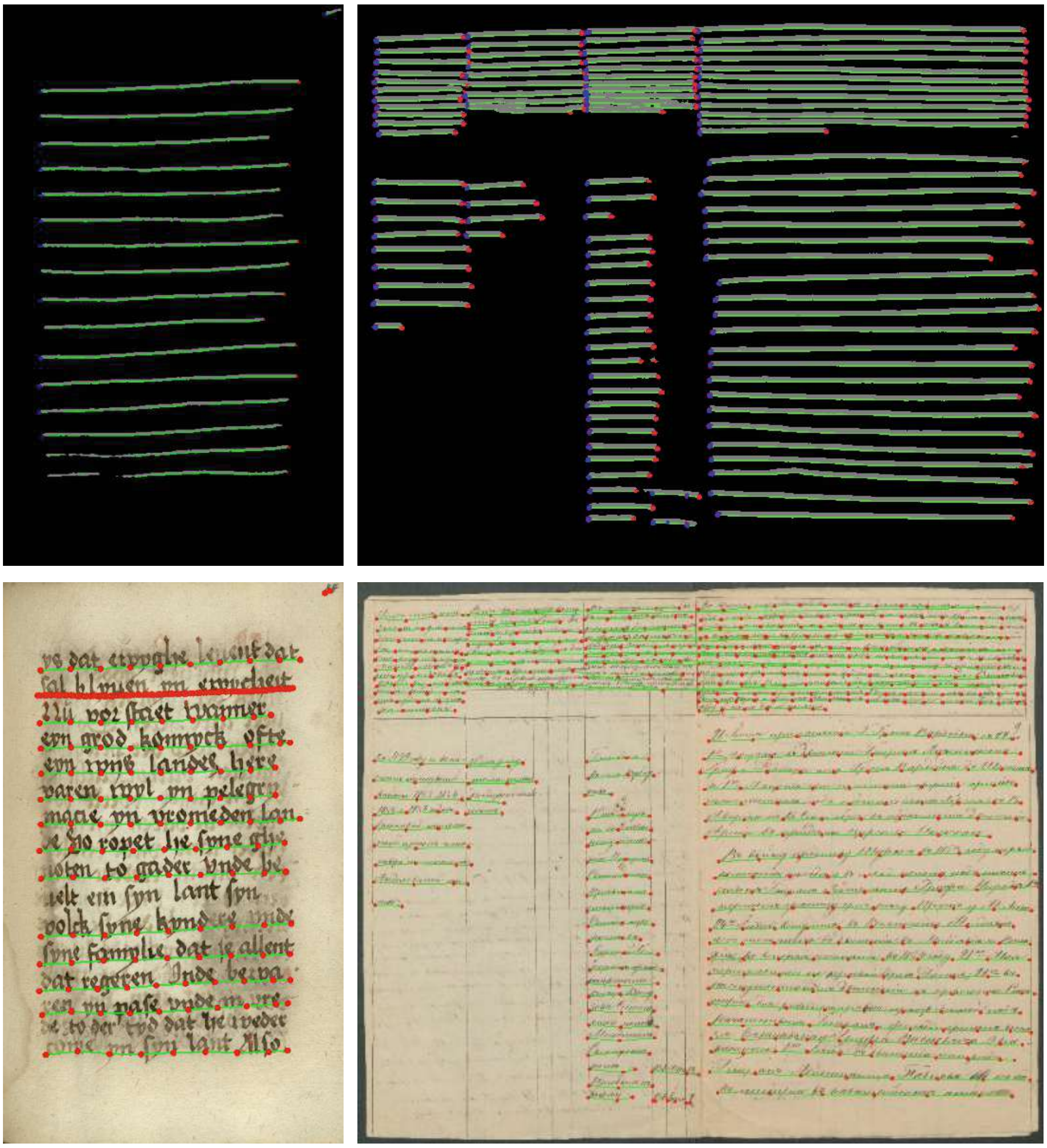


Figure 3.4: A visual exploration of challenges encountered in applying the proposed method on samples from the cBAD 2019 dataset. The top row shows the processed segmentation outputs—denoted by initiation markers in blue, termination points in red, boundary classifications in purple, and text lines in grey—with the original input images complemented by superimposed predicted baselines. The bottom row shows the predicted baselines overlayed onto the input images. Predicted baseline coordinates are pictured with red dots and green lines to visualize the resulting predicted baseline polygons. The left example shows an incorrectly predicted window size and a missed baseline (due to the failure of the segmentation network to identify the starting point correctly). The example on the right illuminates an end-of-line recognition failure, resulting in an erroneous confluence of predicted baselines.

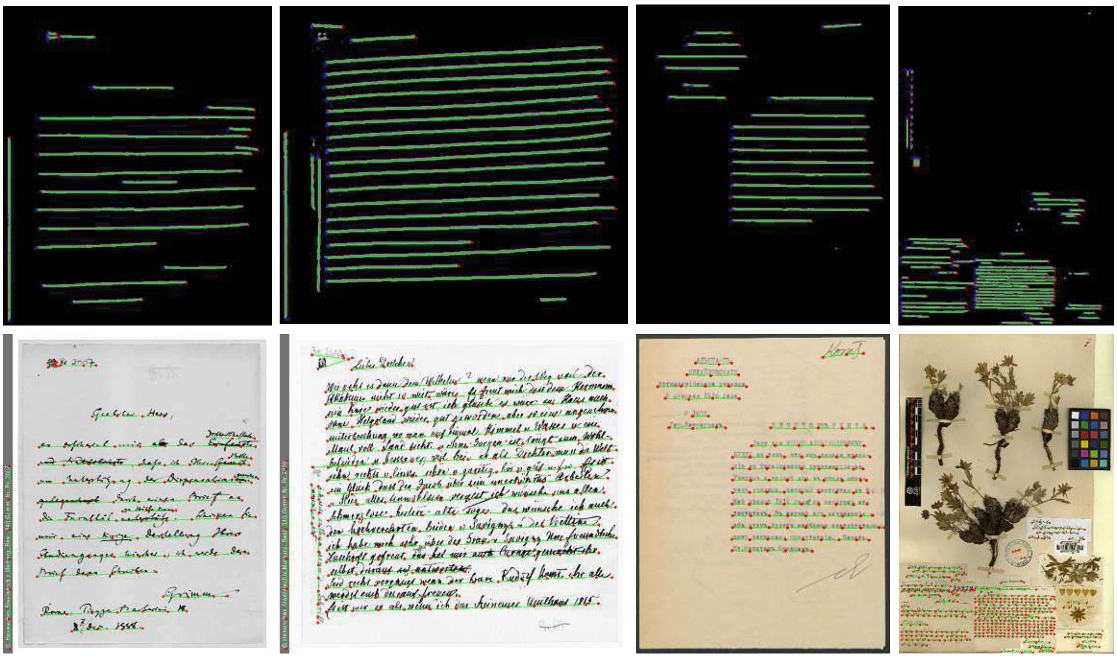


Figure 3.5: Illustrative results of the segmentation model applied to cBAD 2019 dataset test images. The first row visualizes the segmentation model's output: initiation points (blue), termination points (red), boundary class (purple), and baselines (green) with text regions (grey). The second row presents the input image overlaid with the predicted baselines, where red dots represent the predicted baseline coordinates, connected by lines (green).

Name	**Precision**	**Recall**	F_1 **score**
Proposed model	0.872	0.890	0.881
ARU-Net [GLS+19] (Winner cBAD 2019)	0.937	0.926	0.931
Configuration A (With GT start points and angles and with segmentation)	0.953	0.974	0.963
Configuration B (With GT start points and angles and without segmentation)	0.882	0.972	0.924
Configuration C (Without GT start points and angles and without segmentation)	0.788	0.768	0.778

Table 3.3: Results on the cBAD 2019 test dataset. Configurations I and II utilize ground truth points and angles from the test set, which were inaccessible during the cBAD 2019 contest.

are within a tolerance range of the ground truth baselines and counts a predicted point as detected, if so, and undetected otherwise. The resulting scores are measured in terms of precision, recall and F_1 score. The analysis uses two different model configurations:

1. Using only the page images to extract start points and orientations through segmentation, similar to the cBAD 2019 protocol (referred to as the "proposed model" in Table 3.3).
2. Introducing the ground truth initiation points and orientations alongside the page images (as represented by Configuration A and C in Table 3.3) to evaluate the performance of the line rider network independently of the segmentation network.

3.1.3.2 Influence of segmentation on model accuracy

The effect of integrating the segmentation with the raw images on the accuracy of the line rider was further evaluated by dividing the second configuration into Configuration A (using both the raw images and the segmentation masks for baselines and text regions) and Configuration B (using only the raw images). The quantitative results are presented in table 3.3. While the proposed method is outperformed by ARU-Net, the best-performing method on cBAD 2019, the disparity in performance between models with and without ground truth data (*proposed model* vs *Configuration A*) indicates that inaccuracies in the prediction of initiation point and orientation have a significant impact on the overall results. Furthermore, the difference in performance between Configuration A and Configuration B shows that the additional segmentation data helps the line rider to estimate the endpoints of the baselines accurately.

3.1.3.3 Qualitative Insights

Figure 3.5 shows predictions for a selection of typical document images from the cBAD 2019 test set. The proposed method is able to correctly predict baselines in a wide variety of settings, layouts, writing styles and orientations. Figure 3.4 shows two failure cases of the proposed method. Typical errors are undetected baseline start points, incorrect window size and overlapping baselines due to undetected baseline endpoints.

3.1.4 Conclusion

In this section, a novel model that predicts text baselines with a recurrent CNN was presented. The model is based on a visual attention mechanism to follow the text and predicts the baseline coordinates in a recurrent manner, following the line sequentially. The method can be used as long as the starting points of the baselines are known or in a stand-alone manner using the proposed segmentation model. The model can be trained end-to-end using document images with text baselines marked as polygons. An analysis of the errors in the proposed pipeline compared to those of the method proposed by Gruning et al. [GLS+19], as shown in table 3.3, shows that inaccuracies are mainly due to the misdetection or non-detection of start points. This is particularly evident for documents with start points close together, resulting in overlapping predictions. Ambiguity in the location of the start-point further amplifies these detection challenges, as illustrated in Figure 3.4. The model requires little training and converges after only 5 epochs on the 755 images of the cBAD 2019 training set due to the compact structure of the CNNs. The time required for inference on a document depends on the number of baselines and the window size chosen, resulting in variability that contrasts with more uniform methods such as those described in [GLS+19].

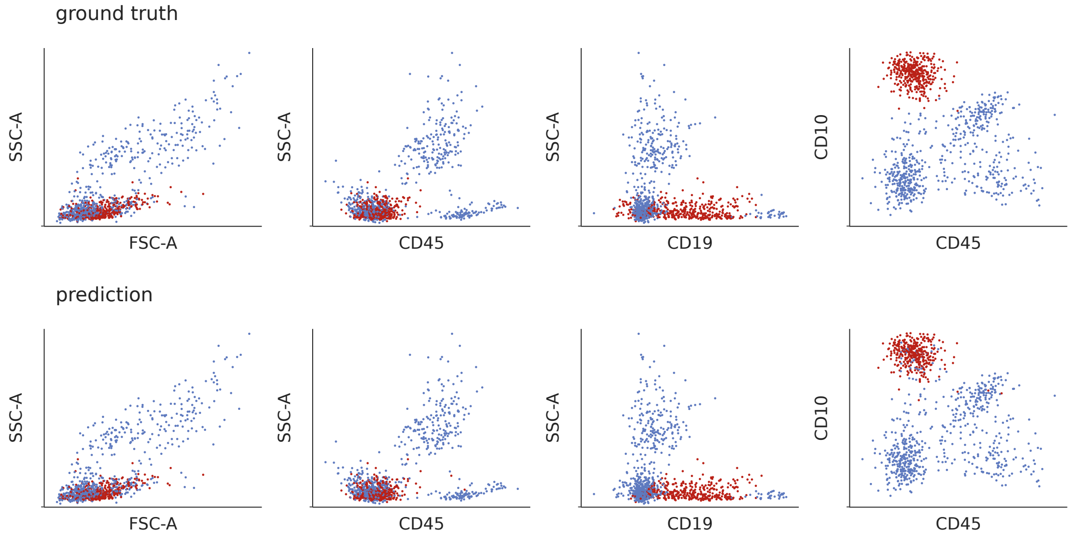


Figure 3.6: An example prediction (bottom row) of the proposed method and the corresponding manual labelling (top row). Red dots denote blast, and blue dots denote non-blast cells. Every plot shows a different 2-dimensional projection of the same underlying FCM data sample. For this visualization, 5000 cells are randomly sampled from a single sample of the *bue* dataset. The prediction is from a model trained on *vie14* (see Table 3.4 for a description of the datasets).

3.2 Modelling Flow Cytometry Data with Global Attention

Acute Lymphoblastic Leukaemia (ALL) is a malignant disorder of lymphoid progenitor cells. It is the most common haematological malignancy in children and adolescents with relapse rates of treated patients of $15 - 20\%$ [PRL08]. Minimal residual disease (MRD) - the proportion of leukaemic cells (*blast* cells) remaining after treatment - is one way of monitoring the progress of treatment. Low MRD levels early in treatment have been shown to be strong predictors of better outcomes [Cam10]. For this reason, the correct assessment of MRD levels is an integral part of modern treatment. However, the detection of blast cells in a sample often depends on the global structure of the sample, requiring a non-local method that provides a holistic understanding of the sample [RDS+19].

Multiparameter Flow Cytometry (FCM) is a reliable method for obtaining estimates of MRD levels during treatment [DFP+02]. It involves staining a sample of a patient's blood or bone marrow with a specific combination of fluorescence-labelled antibodies that bind to cell antigens. In the flow cytometer machine, the cells are then illuminated by a selection of lasers that allow the detection and measurement of physical properties (granularity, size) and biological properties by detecting the antibodies as they bind to the respective antigens. The resulting data for a single cell (an *event*) is a collection of measurements of cell surface marker concentrations (see Figure 3.6 for an example of FCM data as seen during clinical routine). However, manual analysis of FCM data is time-consuming, subjective, and dependent on the operator's experience, which motivates the need for an automated approach.

Contribution To tackle the shortcomings of manual gating, several methods have been proposed that allow automated FCM analysis. However, the structure of FCM data samples proves to be challenging for neural network-based approaches as these often require data points on a grid (e.g. convolutional neural networks for a 2d grid or recurrent neural networks for sequences). Some methods [SLR$^+$19, LSR$^+$18] circumvent this problem by applying neural networks on single cells instead of samples; however, these approaches can only learn static decision boundaries and cannot capture global sample information. In this section, a novel method is presented for the detection of blast cells and MRD quantification, which is capable of capturing long-range information in the entire data space by attending to all events in a sample at once. The method is a variant of the Transformer [VSP$^+$17b, LLK$^+$19a] model and can detect blast clusters in these high-dimensional samples, which requires a holistic understanding of the samples. The code is available on GitHub (https://github.com/mwoedlinger/flowformer) and a pretrained model is also available via the huggingface library (https://huggingface.co/matth/flowformer).

The remainder of this Section is structured as follows: After a discussion of the related work in the next Section 3.2.1, the method is presented in Section 3.2.2, and the results are discussed in Section 3.2.3.

Note: Parts of this Section are based on work previously published in the Journal for Computers in Biology and Medicine [WRW$^+$22a]. The text has been adapted and, in some cases, except mentioned otherwise, directly reproduced to maintain the precision and specificity of the original work. All figures, except otherwise indicated, are taken without modification from the cited paper.

3.2.1 Related Work

Manual gating methods identify cells based on 2-dimensional projections of the (higher-dimensional) FCM data. Automated methods, on the other hand, can exploit the entire parameter space. Typically, these methods aim to assign the correct population to each individual cell. This produces an output similar to manual gating. This output can be used directly in clinical routine (e.g. for MRD quantification) or as a starting point for further data analysis.

Several works formulate the automated FCM analysis as an unsupervised learning problem, using nonparametric density estimation or clustering methods [SBD$^+$15, AFH$^+$13]. One line of research that has recently shown promising results in both unsupervised [NDR$^+$14, DAYR14, JWF16] and supervised [RRK$^+$16, RDS$^+$19] settings is Gaussian Mixture Models (GMM). In SWIFT [NDR$^+$14], the conventional GMM algorithm is adapted to detect rare sub-populations better; BayesFlow [JWF16] uses a hierarchical Bayesian model where expert knowledge can be incorporated through informative priors. [RRK+16] accounts for inter-sample variation with a supervised approach where GMMs are matched to GMMs of a labelled reference dataset. The method is further developed in [RDS+19], where a closed-form optimization is introduced into the fitting process. Deep learning has been successfully applied to the automated processing of image cell data [IAB+21, IAB+21]. However, apart from these imaging FCM applications [NDBS20, EKB+17, LCD+18], there are few examples of successful application of deep neural networks to FCM data. In [LSR+18, SLR+19, LSS+17], neural networks based on fully connected layers that work on single events are presented. However, these methods can only learn fixed decision boundaries to separate biologically meaningful subpopulations. More recently, a method has been proposed in [ZMH+20] that transforms FCM data into image space and processes it with a trained CNN.

3.2.2 Methodology

This section begins with a brief discussion of the structure of FCM data, followed by a detailed description of the network architecture.

FCM Data A single sample is represented by a matrix $E \in \mathbb{R}^{N \times m}$ (the *event matrix*), where N is the number of cells in the sample (typically $10^5 - 10^6$, the exact value for N is different for individual samples) and m is the number of markers (typically $10 - 20$ for the datasets studied in this thesis, the exact number depends on the antibodies used). The number of cells N and markers m can vary between samples. To ensure consistency between samples from other centers and simplify the architecture, the markers are restricted to a *base panel* of markers present in each sample, i.e. m is kept fixed, and non-base panel marker measurements are discarded during training and testing. Follow-up work investigates ways to extend this method to arbitrary marker combinations[WKR+24]. For each index $n \in \{1, \dots, N\}$, $E_n \in \mathbb{R}^m$ is a quantitative representation of the surface markers on cell n. Ignoring the ordering of cells induced by the FCM machine (i.e. samples are represented as a set of vectors instead of a sequence), a sample can also be viewed as a bag of features (where a feature is the marker measurement vector for a single cell).

3.2.2.1 Network Architecture

The underlying data is unstructured, i.e. not on a low-dimensional grid such as images (2D grid) or text (1D grid). It can be thought of as a kind of point cloud in a higher dimensional space (the number of markers and, therefore, the dimensionality of the events is typically $10 - 20$ in the datasets considered in this thesis) that lacks the symmetries often found in applications in 3D Euclidean space. Attention-based methods such as the Transformer [VSP+17a] are in principle capable of handling such data, but the memory requirements of these models grow quadratically with the size of the input [KVPF20], preventing a naive application to FCM data. To circumvent this problem, a variation of the standard Transformer block can be considered: In Lee et al. [LLK+19a], the standard multi-head self-attention block is replaced by a two-step procedure. For a given input set $X \in \mathbb{R}^{N \times m}$ and $k \in \mathbb{N}$ and a learned set of parameters $I \in \mathbb{R}^{k \times h}$:

1. Latent features $h \in \mathbb{R}^{k \times h}$ are extracted by performing an attention operation between the set of learnable parameters $I \in \mathbb{R}^{k \times h}$ as query and the input set X as key and value input.
2. The resulting hidden features h are used as key and value input for a second attention computation with the input X as query.

This *Induced Attention Block* (IAB) breaks the original $\mathcal{O}(N^2)$ operation into two $\mathcal{O}(N \cdot k)$ operations, which avoids the problem of quadratic complexity (with $k \ll N$ held constant). The latent features can capture global sample information, and the full operation is permutation equivariant [LLK+19a], justifying its application to set data. It should be pointed out that the network as a whole is permutation equivariant, not invariant, i.e. the order of samples in input and output is identical, but the actual computation is independent of this ordering. This allows the identification of output and input positions and, therefore, training with a simple binary classification loss. Using the multi-head attention block from [VSP+17b]

$$\mathrm{MHA}(X, Y) = \mathrm{LayerNorm}(X + \mathrm{A}(X, Y, Y)) \tag{3.2}$$

with the Layernorm from [BKH16] and the attention operation A (with d being the embedding dimension)

$$\mathrm{A}(X, Y, Z) = \mathrm{softmax}(\frac{XY^\top}{\sqrt{d}})Z. \tag{3.3}$$

The MHA is then combined with a row-wise feed-forward layer rFF to create a *Multi-head Attention Block* (MHAB)

$$\mathrm{MHAB}(X, Y) = \mathrm{LayerNorm}(H + \mathrm{rFF}(H)) \tag{3.4}$$

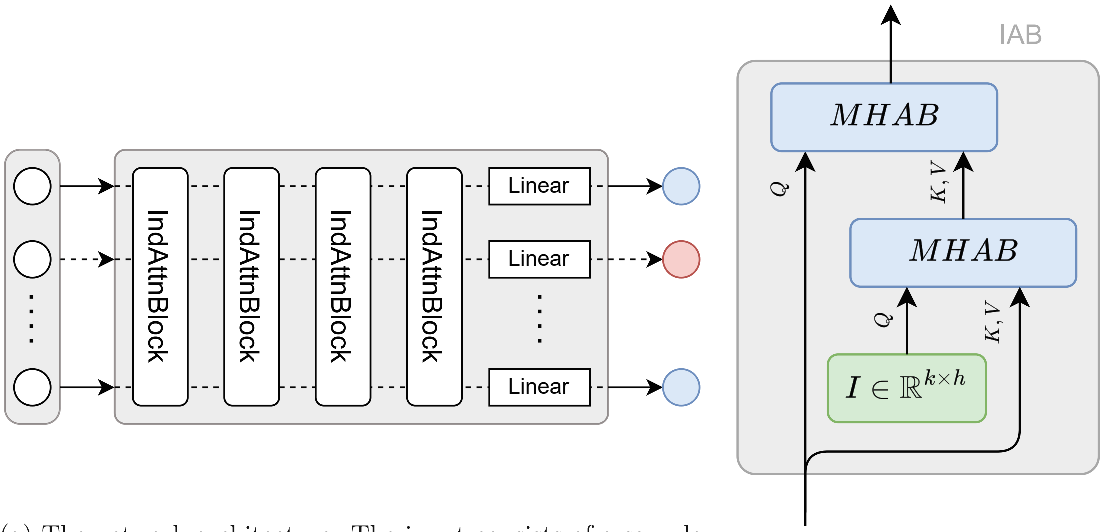
$$\text{where } H = \mathrm{LayerNorm}(X + \mathrm{MHA}(X, Y, Y)). \tag{3.5}$$

Given a set of induced points I, the Induced Attention Block IAB can then be defined as a sequence of MHABs, the first performing cross-attention between the induced points and the input and the second performing cross-attention between the input and the output of the first block

$$\mathrm{IAB}(X) = \mathrm{MHAB}(X, \mathrm{MHAB}(I, X)). \tag{3.6}$$

[2] Using the Induced Attention Block as a building block, one can define a novel neural network that processes a sample of FCM data in a single forward pass.

[2] The first Transformer block can be understood as a *HopfieldPooling* layer from [RSL+20] while the second block performs the computation of the Layer *Hopfield*.



(a) The network architecture. The input consists of a sample represented by the event matrix. For every input cell, a binary classification label is predicted (indicated by the colours blue and red).

(b) The induced attention block from eq. (3.6) as introduced in [LLK+19a] with the learnable parameters I in green and the TransformerBlock from eq. (3.4) in blue.

Figure 3.7: Architecture of the FCM Transformer. Adapted from Wödlinger et al. [WRW+22a].

The network (see Figure 3.7, a) is defined as a sequence of four IABs with a row-wise linear classification layer head, trained with binary cross-entropy loss. Unlike other Transformer models designed for images or text, the raw marker measurements are directly fed into the model without any positional coding. The number of induced points is set to $m = 16$, the latent embedding dimension to $d = 32$ and the number of attention heads to 4 for all three layers.

The resulting model, called *Flowformer* in the results section, is comparatively lightweight with only $\approx 28k$ parameters and can process ≈ 150 samples/s on an NVIDIA GeForce Titan X[3].

Flowformer++ This paragraph describes *Flowformer++*, an improvement of the original flowformer model in multiple aspects, which achieves superior performance for most experiments. The Flowformer++ model has not yet been published outside this thesis.

The induced attention layer described above consists of two MHAB submodules, each containing a single row-wise feed-forward layer. Geva et al. [GSBL20] show that having

[3] Only counting the forward pass of the model, i.e. ignoring the time needed for data loading.

Table 3.4: Description of the FCM datasets used for experiments.

Name	City	Years	#
vie14	Vienna	2009-2014	200
vie20	Vienna	2015-2020	319
vie	Vienna	2009-2020	519
bln	Berlin	2015	72
bue	Buenos Aires	2016-2017	65

two consecutive feed-forward layers can act as a learned key-value storage, improving the knowledge capabilities of these models. Motivated by these findings, the ISAB blocks in the Flowformer++ model contain two successive feed-forward layers separated by a ReLU for each rFF layer in the original Flowformer model. This essentially doubles the feedforward layers compared to the Flowformer model. These additional dense layers increase the number of parameters to $\approx 93k$. In addition, affine transformations and Gaussian resampling of the data points are applied as data augmentation during training. Besides these modifications, the architecture is identical to the original Flowformer model, i.e. it is an encoder-only-Transformer with ISAB blocks instead of vanilla Transformer blocks.

3.2.3 Evaluation

This section begins with a brief discussion of the data in Section 3.2.3.1 and the training in Section 3.2.3.2, followed by the evaluation in Section 3.2.3.3.

3.2.3.1 Data

The proposed method is evaluated on publicly available data[4] from three different clinical centers as well as one internal dataset (vie20). The data consists of bone marrow samples from paediatric patients diagnosed with B-ALL on day 15 after induction therapy. For all samples, ground truth information is available for blast and non-blast cells obtained by manual gating. Table 3.4 provides an overview of the data sets.

Vienna The Vienna dataset was collected at the St. Anna Children's Cancer Research Institute (CCRI) between 2009 and 2020 using an LSR II flow cytometer (Becton Dickinson, San Jose, CA) and FACSDiva v6.2. The dataset is labelled *vie* and contains 519 samples. The Vienna samples are split into two disjunct datasets:

- *vie14:* This dataset contains 200 samples collected between 2009 and 2014. It is identical to the *vie* dataset in [RDS+19]. The samples were stained using a conventional seven-colour drop-in panel ("B7") consisting of the following liquid

[4] flowrepository.org

fluorescent reagents: CD20-FITC/ CD10-PE/ CD45-PerCP/ CD34-PE-Cy7/ CD19-APC/ CD38-Alexa-Fluor700 and SYTO 41.

- *vie20:* This dataset contains 319 samples collected between 2016 and 2020. The samples were stained using dried format tubes (DuraClone™, "ReALB") consisting of the fluorochrome conjugated antibodies CD58-FITC/ CD34-ECD/ CD10-PC5.5/ CD19-PC7/ CD38-APC-Alexa700/ CD20-APC-Alexa750/ CD45-Krome Orange plus drop-in SYTO 41.

Berlin The bln Dura [RDS+19] (hereafter referred to as *bln*) dataset contains 72 samples collected at the Charité Berlin in 2016. These samples were acquired on a Navios flow cytometer (Beckmann Coulter, Brea, CA) and assessed by 8-colour multiparameter FCM ("B8") using a customized dried format tube (DuraClone™, Beckmann Coulter) consisting of the seven fluorochrome-conjugated antibodies CD58, FITC/CD10, PE/CD34, PerCPCy5. 5/CD19, PC7/CD38, APC/CD20, APC-Alexa750/CD45, Krome Orange plus drop-in SYTO 41.

Buenos Aires The bue Dura [RDS+19] (hereafter referred to as *bue*) dataset consists of 65 samples collected at the Garrahan Hospital in Buenos Aires between 2016 and 2017. The staining panel is identical to the bln Dura set (based on the DuraCloneTM cocktail tube, "B8"). Data were acquired on a FACSCanto II (Becton Dickinson, San Jose, CA) using FACSDiva v8.0.1.

3.2.3.2 Training

The proposed method is evaluated for cross-platform compatibility by training separate models for each of the four datasets discussed in the subsection above and then testing these models on each other dataset (except *vie*, as it is only a combination of *vie14* and *vie20*). This results in 12 experiments. In addition, the model is evaluated on a random train/test split of the combined *vie* dataset, giving a total of 13 experiments. Additional experiments are provided to show that the proposed method can be trained on as few as 10 samples and still achieve competitive results. For these experiments, only 10 samples are used for validation.

Flowformer: The Flowformer model is trained using the Adam optimiser [KB14] with an initial learning rate of $1e-3$ and a cosine annealing scheduler [LH16] with 10 iterations and a minimum learning rate of $2e-4$. It is trained for 100 epochs with a batch size of 1 and evaluated on the test with the best checkpoint as measured by the average F_1 score on the validation set.

Flowformer++: The Flowformer++ model is trained for 180 epochs using the AdamW optimizer and a cosine annealing learning rate scheduler.

Table 3.5: The experimental results evaluated with precision (p), recall (r), average F_1-score (avg F_1) and median F_1-score (med F_1). The method is compared to Reiter et al. [RDS+19]. Boldface values indicate the best-performing method for a specific train/test dataset combination.

train	**test**	**p**	**r**	**avg F_1**	**med F_1**	**med F_1**	**med F_1**
					Flowformer	Flowformer++	[RDS+19]
vie	vie	0.81	0.83	0.81	0.94	**0.96**	-
	bue	0.63	0.84	0.66	**0.87**	0.83	*0.68*
bln	vie14	0.77	0.83	0.77	**0.90**	0.89	*0.35*
	vie20	0.79	0.77	0.74	0.87	**0.89**	*0.48*
	bln	0.56	0.92	0.62	0.77	**0.90**	*0.50*
bue	vie14	0.76	0.88	0.79	0.90	**0.93**	*0.84*
	vie20	0.79	0.74	0.72	0.88	**0.93**	*0.86*
	bln	0.78	0.82	0.75	0.90	**0.94**	*0.81*
vie14	bue	0.82	0.81	0.78	0.95	**0.98**	*0.84*
	vie20	0.81	0.74	0.73	0.89	**0.94**	*0.86*
	bln	0.64	0.87	0.66	0.81	**0.95**	*0.25*
vie20	bue	0.82	0.69	0.71	0.86	**0.96**	*0.81*
	vie14	0.82	0.69	0.84	**0.95**	**0.95**	*0.89*

3.2.3.3 Results

The experimental results are listed in Table 3.5. The quality of the results is assessed in terms of average precision (p), average recall (r), average F_1 scores (avg F_1) and median F_1 scores (med F_1) where blasts are considered "positive" and non-blasts are "negative". For samples with no blasts or very few blasts (see the leftmost region in Figure 3.10, especially the first 10 samples where no blasts are present), the F_1 score is not a good measure of performance because misclassification of individual cells can have a significant effect on the F_1 score that is not reflected in the clinical significance. In particular, for samples with zero blasts, misclassifying a cell as a blast cell will result in a F_1 score of 0 while the MRD is ≈ 0, making such a single false prediction clinically insignificant while having a significant impact on the average F_1 score. For these reasons, the median F_1 score is preferred to the average F_1 score for measuring model performance.

The method is compared with the GMM-based model described in Reiter et al. [RDS+19], which is evaluated on the vie14, vie20, bln and bue datasets. The complete set of results for the experiments performed can be seen in Table 3.5. The existing approach [RDS+19] is outperformed in all experiments. The proposed method is significantly faster with inference times of 5ms versus 3000ms for the GMM-based approach [RDS+19]. For bue/bln in particular, the Flowformer model only achieves a median F_1 score of 0.77 with a precision of 0.56 and a recall of 0.92, suggesting that performance may degrade

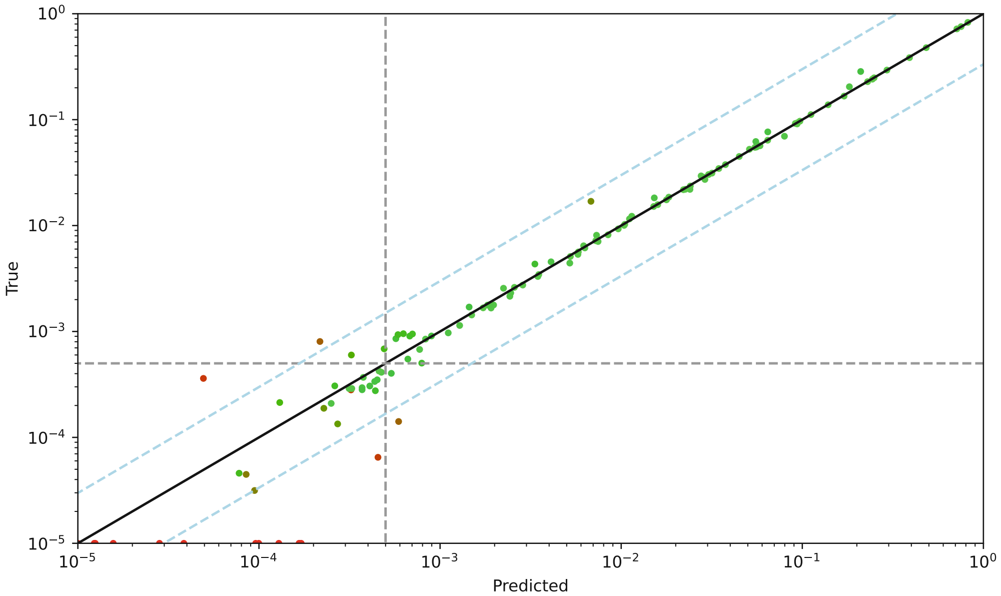


Figure 3.8: F_1 scores and predicted MRD values for the test set of the vie experiments. Each point represents a single sample, with the colour indicating the F_1 score (colours going from red = 0.0 to green = 1.0). The dashed lines correspond to MRD values of $5e-4$, which is the minimal resolution required for patient stratification according to the current international therapy trials of the allied study groups of the iBFM consortium. Predictions that are either less than 3 times or more than 1/3 of the true MRD are considered acceptable (correct) predictions. [DGR+08].

for sufficiently different data sources. However, adding 5 random samples from the test set to the training set and testing on the remaining samples improves the median F_1 score, precision and recall to 0.87, 0.7 and 0.91, respectively, indicating that when a small amount of labelled data is available, the cross-laboratory performance of the flowformer method can be significantly improved. In general, it can be seen that the proposed method performs better for samples with larger MRD values when measured in terms of F_1. Figure 3.9 shows the average F_1 score for all samples with an MRD value above the threshold given by the value on the x-axis for the vie test set. Samples with a low F_1 score predominantly have a lower MRD, i.e. a lower number of blast cells. For low MRD values, the flowformer model tends to overestimate the true value more often than it underestimates it. This can be seen in Figure 3.8, where the ground truth MRD is plotted against the predicted MRD. A different visualization is given in Figure 3.10 where the true MRD, the predicted MRD and the F_1 score are given for each sample. The Flowformer++ model performs better for every experiment except bln/bue and bln/vie14. See Appendix A for additional MRD plots, calibration histograms for the Flowformer++ model for each experiment, and a more extensive comparison between the Flowformer and Flowformer++ models.

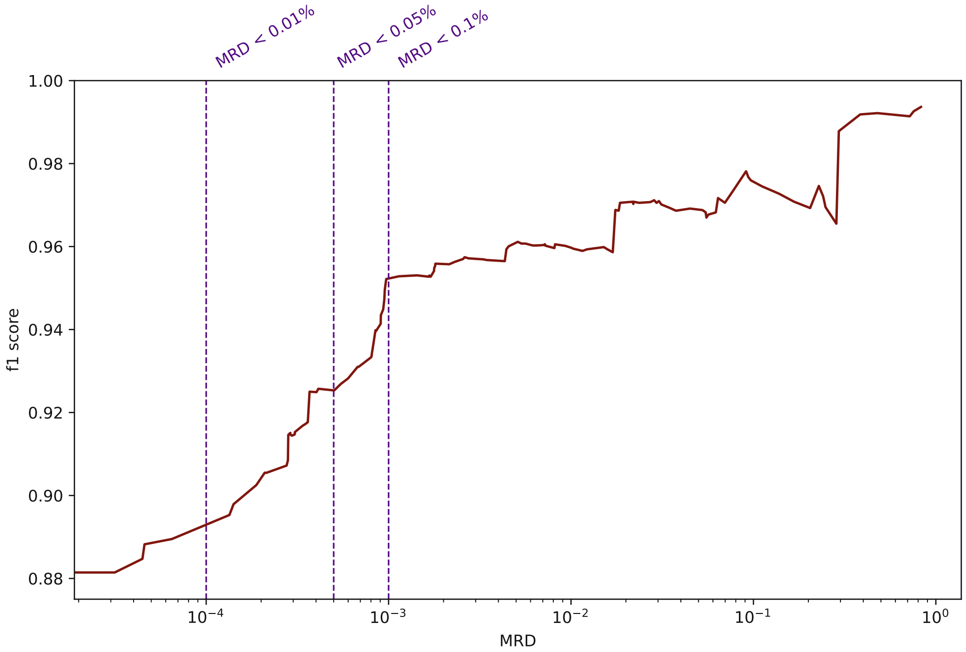


Figure 3.9: Running average of F_1 scores against ground truth MRD values, i.e. for a given MRD value x, the line shows the average of F_1 scores of all samples within the vie test set with an MRD value greater than or equal to x. Due to the logarithmic scale, the 11 samples in the vie test set with MRD values of 0 are not shown, which explains the mismatch between the lowest running average of 0.88 and the mean across the test set of 0.81.

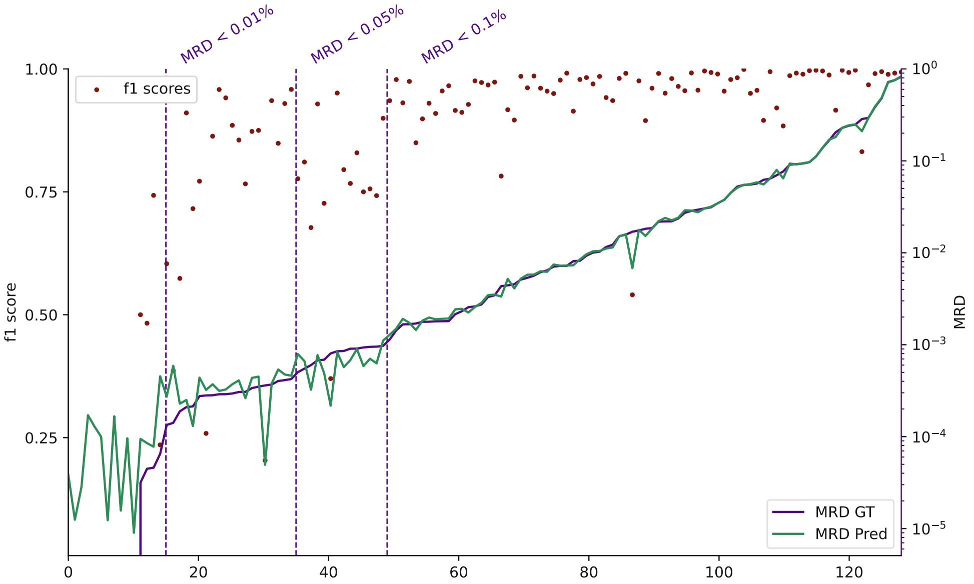


Figure 3.10: F_1 scores (red dots), ground truth MRD values (purple lines) and predicted MRD values (green line) for the vie test set.

3.2.4 Conclusion

In this section, a novel method for automated cell population identification is proposed and trained for blast cell detection in B-ALL FCM data. The method is based on a lightweight neural network that allows fast (≈ 150 samples/s) processing of samples with $10^5 - 10^6$ cells on an NVIDIA GeForce GTX TITAN X. It is trained in a supervised manner on as few as 65 samples of data from three different sources, while still being capable of generalizing to unseen data. The method differs from existing approaches that utilize neural networks for automated FCM analysis [SLR+19, LSR+18] in that it relies heavily on attention. This allows the network to model the holistic properties of samples rather than relying solely on local features as in existing work [LSR+18]. The quadratic complexity due to the long range of the attention operation is circumvented by using a SetTransformer[LLK+19a] inspired architecture, where the original $\mathcal{O}(n^2)$ complexity is replaced by two operations of $\mathcal{O}(nk)$, where $k \ll n$.

3.3 Modelling Stereo Image Redundancies with Stereo Cross Attention

Stereo Image Compression (SIC) aims to compress the two images of a stereo image pair more efficiently by exploiting their mutual information. This process is similar to the lossy compression of individual images, aiming to decrease the storage and transmission bit rate while preserving the content integrity and perceptual quality. Stereo cameras, particularly in applications such as autonomous driving and virtual reality streaming, require high compression rates and low latency in encoding and decoding for seamless recording or streaming. In this context, learned compression techniques, characterized by their symmetric encoding and decoding durations, offer an advantage over traditional methods where fast decoding is often achieved at the expense of slow encoding. These learned methods convert images into a latent space representation, which is then quantized and encoded into a bitstream using a learned probability distribution, followed by entropy coding. They are trained end-to-end to optimize the balance between bitrate and distortion, to minimize the cross-entropy between the actual latent distribution and its estimated model. Learned compression methods have recently surpassed conventional techniques such as BPG [Bel14] in single image compression [MBT18]. Given the substantial mutual information in stereo image pairs, an ideal method for compressing stereo images should approach the bitrate of one image in the pair, significantly outperforming the independent compression of each image. However, the challenges posed by occlusions and the different fields of view in stereo setups make it difficult to achieve significant bitrate reductions. Early learned compression models for stereo image data focused on modelling the disparity between the image pairs to create a dense warp field [LWU19a] or applying a rigid homography transform [DYY$^+$21] to support image registration to exploit their similarities. However, these methods are computationally expensive and inefficient for transmitting dense disparity maps through the bitstream. Furthermore, incorrect or noisy disparity map predictions can increase the transmitted images' bitrate by introducing additional noise into the latent. Neural attention is a promising idea that could help to overcome these problems. A neural attention connection between the left and right image streams allows for input-specific information transfer between the left and right image streams. Furthermore, if one restricts oneself to rectified images, the problem of the quadratic complexity of neural attention becomes tractable if attention itself is restricted to the epipolar line. The use of such a stereo cross-attentional connection allows the learned modelling of stereo redundancies without explicit computation of any form of warping. In this thesis, two methods are described that implement two types of stereo image attention and achieve SOTA performance at the time of publication:

Stereo Attention for Stereo Image Compression (SASIC): In the SASIC model, the left image is encoded with the encoder and hyperprior encoder modules, similar to single image encoding. After the encoder has processed the right image, the optimal horizontal shifts (minimizing the mean square error) are determined for each channel of its latent representation concerning the corresponding channel of the left image latent, and the two shifted channels are subtracted so that only the residual is encoded for the right latent. This is motivated by the observation that the dominant rigid transformation between rectified images in a stereo pair is a horizontal shift, and working in latent space results in a larger effective disparity range for a given shift due to downsampling. In addition, a stereo attention module is used in the decoder to connect the left and right image streams and allow the model to accurately decode the joint latent. The code is available on GitHub: `https://github.com/mwoedlinger/sasic`.

Epipolar Cross Attention for Stereo Image Compression (ECSIC): The ECSIC model for stereo image compression omits explicit disparity estimation. The network is structured as an autoencoder and incorporates a hyperprior [BMS+18] based entropy model. It features a unique stereo attention module within the encoder and decoder, which allows simultaneous processing of both images in the stereo pair. The novel stereo-attention module connects the left and right streams via a cross-attention connection. Attention is restricted to the epipolar line, thus avoiding the problems of quadratic complexity of the attention operation. Additionally, two stereo context modules are implemented within the entropy model, which improves the estimation accuracy by using the left image as a reference for the right image. The code is available on GitHub: `https://github.com/mwoedlinger/ecsic`.

The remainder of this section is structured as follows. In Section 3.3.1, a brief review of the existing literature on image compression (traditional/learned/stereo) is given; in Section 3.3.2, the SASIC and ECSIC models are described. The results and ablation studies are discussed in Section 3.3.3, and a discussion in Section 3.3.4 concludes the section.

Note: This section is based on previously published work. The SASIC model was published in the Proceedings of the Conference for Computer Vision and Pattern Recognition (CVPR) 2022 [WKXS22], and the ECSIC was published in the Proceedings of the Winter Conference on the Applications of Computer Vision 2024 [WKK+24]. The text has been adapted and, in some cases (including the introduction to this section), directly replicated to maintain the precision and specificity of the original work. In particular, the method sections are taken directly from the original publications. All figures are taken without modification from the cited paper.

3.3.1 Related Work

Modern image compression techniques can be broadly grouped into two main categories: traditional and learned. Traditional methods rely on manually designed transformations of the input image into its latent form. In contrast, learned methods use data-driven optimization of the rate-distortion loss to learn the transformation. Both methods use an entropy coder to transform the discrete latent representation to and from a minimum length bitstream and vice versa, and an approximate inverse transform is used for image reconstruction.

Traditional Methods The Joint Photographic Experts Group (JPEG) created the JPEG standard in 1992 [Wal92]. The image codec includes a fixed 8x8 block tiling, chroma subsampling, a discrete cosine transform, and multiple prediction modes for subsequent blocks. Its successor, JPEG2000 [SCE01], introduces multi-resolution processing using a discrete wavelet transform. Recent advances in compression technology have been primarily in video, with modern image codecs typically introduced as intra-frame compression in modern video codecs, including BPG [Bel14] (derived from HEVC [SOHW12]), AVIF [All22] (based on AV1), and VVC-intra [BWY+21]. Although VVC-intra offers superior compression efficiency among traditional codecs, its practical adoption is hampered by its slow encoding, the lack of readily available production-ready decoders, and restrictive licensing issues.

Learned Methods Initial breakthroughs in learned image compression were made by Toderici et al. [TOH+16], who developed a recurrent neural network-based method for variable-rate image compression. Ballé et al. [BLS17] further advanced the field by introducing an autoencoder-based model that trains with a rate-distortion loss for a given target bitrate, using a fixed parameterized latent distribution for the entropy model. Later work by the same group replaced the fixed latent entropy model with a per-pixel Gaussian distribution, where the parameters are tailored for each input image using a hyperprior module [BMS+18] or an autoregressive context module [MBT18, MAT+18], significantly improving performance. Subsequent improvements to this basic structure of an encoder/decoder augmented with a hyperprior/autoregressive entropy model have been varied, including modifications to the model architecture [CSTK20, GYP+21, XCC21, HYP+22], advances in quantization techniques [TSCH17, GZFC21b], and new theoretical approaches to the optimization challenge [YBM20]. Significant efforts have also been dedicated to refining the context model [MS20, QTS+20, HZS+21, GZFC21a, HYP+22, KGB+22]. Recent innovations show performance improvements by integrating Transformers or other attention mechanisms, mainly in the hyperprior and context model [KGB+22, QSL+22, KHL22], and also in the primary autoencoder [ZYC22, ZSZ22]. A distinct research trajectory focuses on enhancing the realism of reconstructed images alongside the traditional goal of rate-distortion optimization. This is mainly achieved by GANs [TAL18, ATM+19, MTTA20, GSG+21, HYY+22], but also by other generative models such as denoising diffusion models [TSHM22, GPW+23].

Stereo Image Compression Stereo image compression uses the mutual information between the left and right images of a stereo pair to save bitrate. Although similar to video frame compression, the disparity in stereo images is not adequately captured by optical flow, making direct application of video codecs less effective. Among traditional methods, MV-HEVC [MMSW06] extends the HEVC video codec for multi-view sequences, offering robust performance but lacking support for higher bit-depth processing and 444 chroma mode. Huang et al. [HSD+21] introduced a learnable lossless stereo compression method based on explicit disparity estimation and image warping.

Several learned methods have been proposed for learned lossy stereo compression: The DSIC method by Liu et al. [LWU19a] uses a conditional entropy model where features from the first encoded image, warped by disparity, are used to encode the second image. Deng et al. [DYY+21] introduced the HESIC method, where the second image is warped using an estimated homography and only the residual is encoded, complemented by a context-based entropy model and a final quality enhancement module to minimize bitrate and improve reconstruction quality. Zhang et al. [ZSZ23] propose the LDMIC method, which uses decoder-only cross-attention connections for distributed multi-view compression. The encoder module is shared between different views and encodes each separately, while the decoder connects different views with an attention layer. Additionally, an autoregressive entropy model is used for improved entropy coding. Mital et al. [MÖGG23] propose a similar approach for distributed source coding, where a correlated image is available during decoding.

3.3.2 Methodology

Two methods for stereo image compression are presented in this section. The SASIC model [WKXS22] uses cross-attention connections only in the decoder and a channel-wise translation transform (an image shift) as a cheap (i.e. few additional bits) alternative to a full disparity warp in the latent. The ECSIC model [WKK+24] is an improvement over SASIC and includes no disparity modelling heuristics. Instead, the stereo cross-attention module is naturally included in both the encoding and decoding parts of the model. In addition, an improved context module is proposed.

3.3.2.1 SASIC

Figure 3.11 shows an overview of the proposed method. It compresses a stereo image pair into two streams that are connected in the latent entropy model and the decoder. The hyperprior model estimates the parameters of the latent entropy model. For a given stereo image pair $\boldsymbol{x}_1, \boldsymbol{x}_2$, in the first step, the left image is encoded independently from the right image. Then the right image is processed by the encoder module E, and the optimal channel-wise horizontal shift for the quantized left latent $\hat{\boldsymbol{y}}_1$ is computed such that the MSE to the right latent $\boldsymbol{y}_2$ is minimal. For each channel c in the latent representations y_2, the optimal shift $s_c = \operatorname{argmin}_s \operatorname{MSE}\left(\boldsymbol{y}_2^{(c)} - \operatorname{shift}_s(\boldsymbol{y}_1^{(c)})\right)$ is determined, where $\operatorname{shift}_s(\boldsymbol{y})$ is defined as a tensor of the same size as $\boldsymbol{y}$ but shifted horizontally (with respect to the original image) by s pixels (zero-padded if necessary). Instead of $\boldsymbol{y}_2$, only the residuals are encoded, defined for each channel as $\boldsymbol{y}_{\text{res}}^{(c)} = \boldsymbol{y}_2^{(c)} - \operatorname{shift}_c(\boldsymbol{y}_1^{(c)})$. The search range for s_c for the experiments is limited to 64 pixels (in the downsampled latent representation) in one direction only (stereo disparity has only one polarity). The optimal shift can be found efficiently using a convolution in the horizontal direction (achieved by appropriate padding) and element-wise operations to compute the MSE. It is, therefore,

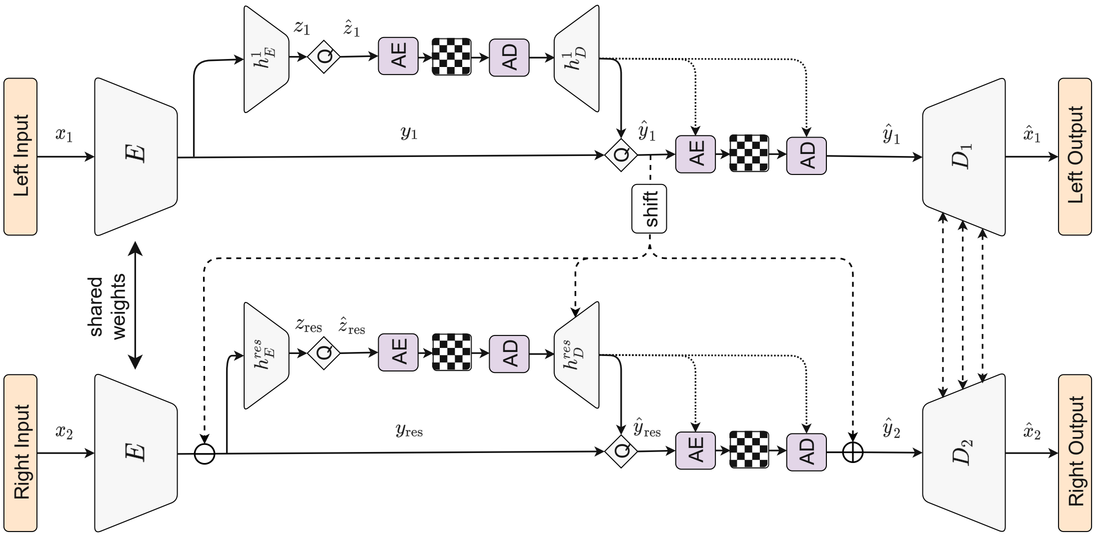


Figure 3.11: The full architecture of the proposed method. The structure of submodules is shown in Figure 3.12 and Figure 3.13. The arithmetic encoder *AE* and the arithmetic decoder *AD* are not relevant during training. The bitstreams are pictured with a checkerboard pattern. Dotted lines are connections that are not relevant during training, and dashed lines show connections between the left and the right sides.

not significantly more demanding than other common operations in CNNs. The residual between the right latent and the shifted left quantized latent

$$\boldsymbol{y}_{\text{res}} \coloneqq E(\boldsymbol{x}_2) - \text{shift}(\hat{\boldsymbol{y}}_1) \tag{3.7}$$

is then encoded. During decoding, the left latent is decoded first and then the shifted left quantized latent $\text{shift}(\hat{y}_1)$ is added to the quantized residual $\hat{y}_{\text{res}}$ to obtain the right latent

$$\hat{\boldsymbol{y}}_2 \coloneqq \hat{\boldsymbol{y}}_{\text{res}} + \text{shift}(\hat{\boldsymbol{y}}_1). \tag{3.8}$$

In the final step, $\hat{\boldsymbol{y}}_1$ and $\hat{\boldsymbol{y}}_2$ are processed jointly in the decoder modules D_1, D_2.

Applying a channel-wise shift is computationally cheap and requires almost no additional side information. Since the encoder performs 4× downsampling, a maximum shift of 64 pixels in the latent corresponds to a shift of 256 pixels in the original image. This equals to 72 bits of side information (6 bits times 12 latent channels), which for a 512×512 input image results in an overhead of only ≈ 0.00027 bits/pp. Furthermore, a simple shift is also theoretically motivated by the fact that for a rectified stereo image pair, a shift describes the transformation between the two image planes.

Encoding modules and quantization The encoder/decoder architecture is loosely based on the single image compression method proposed in [XLC+20]. The encoder

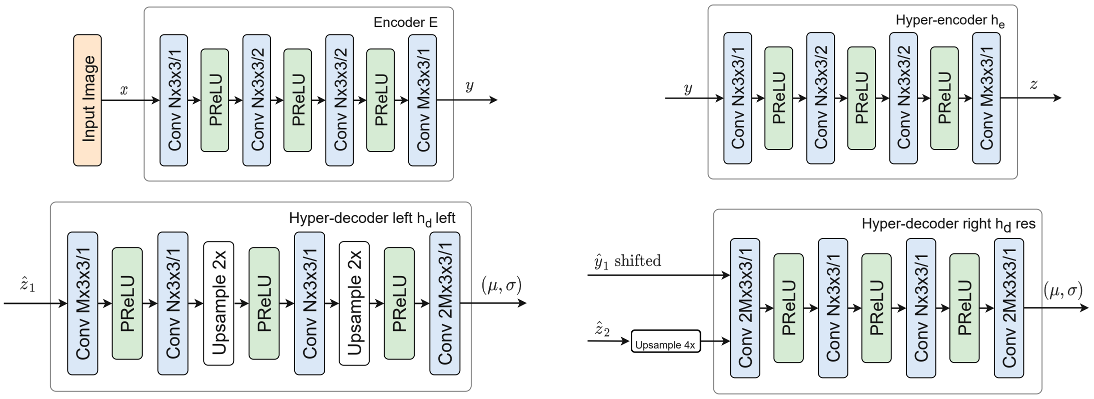


Figure 3.12: The top row shows the architecture of encoder E and hyper-encoder h_E. The bottom row shows the decoder of the hyperprior with the decoder for the left image bottom left and the decoder for the right image bottom right. In all experiments $N = 192$ and $M = 12$

module E and the hyperprior encoders h_E^1 and h_E^{res} each consist of four convolutional layers with Parametrized Rectified Linear Units (PReLUs) [HZRS15] as nonlinearities. The structure of the encoder modules is shown in the top row of Figure 3.12. For both encoder modules, downsampling occurs in the second and third convolution, resulting in a 4× downsampling for the latent and 16× downsampling for the hyperlatent compared to the size of the inputs $\boldsymbol{x}_1, \boldsymbol{x}_2$. The same encoder module E (i.e. shared weights) is used for the left and right images, and the same architecture for h_E^1 and h_E^{res} (with separate weights). Motivated by the discussion in [PFBK21], during training, the noise approximation quantization [BLS17] is used for the rate loss, and a straight-through estimation (STE) quantization is used for the distortion loss.

Decoding The architectures of the hyperprior decoders follow the same general structure of four convolutional layers with PReLUs as nonlinearities; see the bottom row of Figure 3.12. The hyperprior decoder for the left image h_D^1 receives the quantized hyperlatent $\hat{\boldsymbol{z}}_1$ as input and performs nearest neighbour upsampling after the second and third convolutional layers. The hyperprior decoder for the residual h_D^{res} receives both the 4× upsampled quantised hyperlatent $\hat{\boldsymbol{z}}_{res}$ and the shifted $\hat{\boldsymbol{y}}_1$ as input. There is no additional upsampling after the convolutional layers in h_D^{res}. The final decoder modules D_1 and D_2 again consist of four convolutional layers with PReLU activation functions and upsampling after the second and third convolution, but with stereo attention modules (SAM) from [YWW$^+$20] before the first three convolutional layers connecting the left and right decoder streams; see overview in Figure 3.13. SAM works by computing an attention mask between the left and right inputs, which is used to warp left to right and vice versa. The input is stacked with the warped image in the channel dimension and processed by the next convolutional layer. Attention is computed only between positions on the same epipolar line (assuming the images are rectified), which avoids the problem of quadratic complexity in the sequence length of the attention mechanism.

Entropy estimation Optimal entropy estimation is essential for the rate loss term during training and for correct bitrate allocation during testing. For stereo image compression, where a pair of images with mutual information $H(\boldsymbol{x}_1, \boldsymbol{x}_2) > 0$ is compressed together, using the left latent as side information in the entropy model of the residual can in principle reduce the bitrate even further. As discussed in 3.3.2.1, the quantization of the latent and hyperlatent is approximated with noise during training $\tilde{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\epsilon}$ and $\tilde{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{U}(-0.5, 0.5)$ following Balle et al. [BLS17]. During testing, $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{z}}$ are replaced with their integer quantized equivalents $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{z}}$. Similar to [BMS+18], a hyperprior model extracts meta-information $\tilde{\boldsymbol{z}}_n, n \in \{1, 2\}$ to reduce the entropy of the latents $\tilde{\boldsymbol{y}}_n, n \in \{1, 2\}$. Following the discussion in [BMS+18], the probability of the hyperprior $\tilde{\boldsymbol{z}}_n$ is modelled by a convolution of a parametric probability function $q_{\tilde{\boldsymbol{z}}_n}$ and a uniform distribution u.

$$p_{\tilde{\boldsymbol{z}}_n}(\tilde{\boldsymbol{z}}_n \mid \boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_n}) = (q_{\tilde{\boldsymbol{z}}_n} * u)(\tilde{\boldsymbol{z}}_n) \tag{3.9}$$

where $\boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_n}$ denotes the parameters of $q_{\tilde{\boldsymbol{z}}_n}$ and $u(\tau) = \mathbb{1}_{[-0.5,0.5]}(\tau)$. $p_{\tilde{\boldsymbol{z}}_n}(\tilde{\boldsymbol{z}}_n \mid \boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_n})$ can then be expressed via the cumulative density function $F_{\tilde{\boldsymbol{z}}_n}$ of $q_{\tilde{\boldsymbol{z}}_n}$:

$$\begin{aligned} p_{\tilde{\boldsymbol{z}}_n}(\tilde{\boldsymbol{z}}_n \mid \boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_n}) &= \int_{-\infty}^{\infty} q_{\tilde{\boldsymbol{z}}_n}(\tau \mid \theta_{\tilde{\boldsymbol{z}}_n}) \mathbb{1}_{[-0.5,0.5]}(\tilde{\boldsymbol{z}}_n - \tau) d\tau \\ &= F_{\tilde{\boldsymbol{z}}_n}(\tilde{\boldsymbol{z}}_n + 0.5 | \theta_{\tilde{\boldsymbol{z}}_n}) - F_{\tilde{\boldsymbol{z}}_n}(\tilde{\boldsymbol{z}}_n - 0.5 | \theta_{\tilde{\boldsymbol{z}}_n}) \end{aligned}$$

The probability density function of $q_{\tilde{\boldsymbol{z}}_n}$ is modelled as a fully factorized Laplacian distribution

$$\mathrm{Lap}_{\boldsymbol{\mu}_n, \boldsymbol{b}_n}(\tilde{\boldsymbol{z}}_n) = \prod_i \frac{1}{2b_{n;i}} \exp\left(-\frac{|\tilde{z}_{n;i} - \mu_{n;i}|}{b_{n;i}}\right), \tag{3.10}$$

where i denotes the pixel index and the parameters $\mu_{n;i} \in \mathbb{R}, b_{n;i} \in \mathbb{R}^+$ are shared between all positions in a channel. The set of parameters $(\boldsymbol{\mu}_n, \boldsymbol{b}_n)$ are denoted by $\boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_n}$.

The latent distributions are modelled as convolutions of a parametric probability function $q_{\tilde{\boldsymbol{y}}_n}$, with $n \in \{1, \mathrm{res}\}$, and a uniform distribution u.

$$p_{\tilde{\boldsymbol{y}}_1}(\tilde{\boldsymbol{y}}_1 \mid \tilde{\boldsymbol{z}}_1, \boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_1}) = (q_{\tilde{\boldsymbol{y}}_1} * u)(\tilde{\boldsymbol{y}}_1) \tag{3.11}$$

$$p_{\tilde{\boldsymbol{y}}_{\mathrm{res}}}(\tilde{\boldsymbol{y}}_{\mathrm{res}} \mid \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_{\mathrm{res}}, \boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_{\mathrm{res}}}) = (q_{\tilde{\boldsymbol{y}}_{\mathrm{res}}} * u)(\tilde{\boldsymbol{y}}_{\mathrm{res}}) \tag{3.12}$$

The distributions are conditioned on the hyperpriors $\tilde{\boldsymbol{z}}_n$ and the parameters of the hyperprior decoder $\boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_n}$. For the residual latent $\tilde{\boldsymbol{y}}_{\mathrm{res}}$, the probability distribution is additionally conditioned on $\tilde{\boldsymbol{y}}_1$ by using the shifted $\tilde{\boldsymbol{y}}_1$ as an additional input to the decoder of the hyperprior h_D^{res}. The first latent $q_{\tilde{\boldsymbol{y}}_1}$ is modelled as a fully factorized Laplacian and, contrary to the hyperlatents where the learned parameters are shared

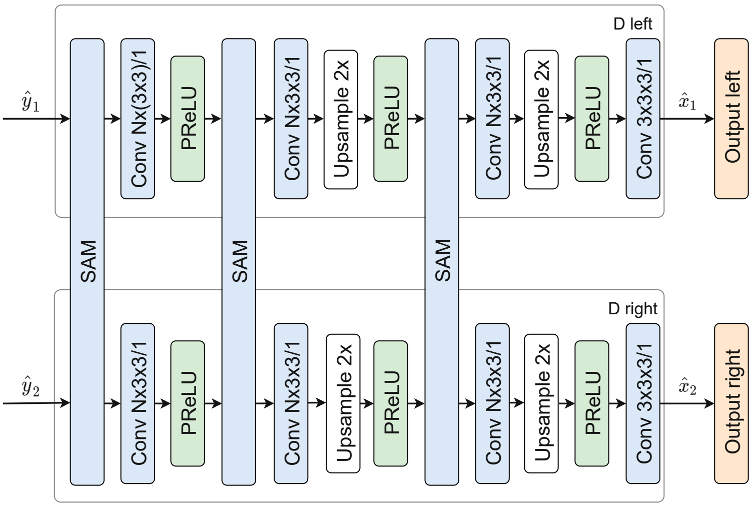


Figure 3.13: The decoder architecture. The SAM blocks contain the stereo attention module proposed in [YWW⁺20]. The number of channels N is set to 192.

for each position for a given channel, a different set of parameters is predicted for each position and channel with

$$\boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_1} = h_D^1(\hat{\boldsymbol{z}}_1) \tag{3.13}$$

$$\boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_{res}} = h_D^{res}(\hat{\boldsymbol{z}}_{res}, \text{shift}(\hat{\boldsymbol{y}}_1)). \tag{3.14}$$

The total rate is then the sum of the cross entropies for $\tilde{\boldsymbol{z}}_1, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_{\text{res}}, \tilde{\boldsymbol{y}}_{\text{res}}$:

$$\begin{aligned} \mathcal{R} = \mathbb{E}_{\boldsymbol{x_1},\boldsymbol{x_2} \sim p_{\boldsymbol{x}}} \big[& -\log_2 p(\tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_1 \mid \boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_1}, \boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_1}) \\ & -\log_2 p(\tilde{\boldsymbol{y}}_{res}, \tilde{\boldsymbol{z}}_{res} \mid \tilde{\boldsymbol{y}}_1, \boldsymbol{\theta}_{\tilde{\boldsymbol{z}}_{res}}, \boldsymbol{\theta}_{\tilde{\boldsymbol{y}}_{res}}) \big], \end{aligned} \tag{3.15}$$

where $p_{\boldsymbol{x}}$ denotes the true distribution of the input data.

Training The model is trained with the rate-distortion loss

$$\mathcal{L} = \mathcal{R} + \lambda \mathcal{D}, \tag{3.16}$$

where $\mathcal{R}$ is the rate term from eq. (3.15) and $\mathcal{D}$ denotes the distortion metric, which is equal to the sum of the MSE values for the left and right images between the inputs $\boldsymbol{x}_1, \boldsymbol{x}_2$ and the predictions $\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{x}}_2$,

$$\mathcal{D} = \mathbb{E}_{\boldsymbol{x}_1,\boldsymbol{x}_2 \sim p_{\boldsymbol{x}}} \big[\, \|\boldsymbol{x}_1 - \hat{\boldsymbol{x}}_1\|^2 + \|\boldsymbol{x}_2 - \hat{\boldsymbol{x}}_2\|^2 \, \big]. \tag{3.17}$$

The model is trained for different values of $\lambda \in \{1\text{e}{-3}, \dots, 4\text{e}{-1}\}$ to achieve different desired target bitrates. For each bitrate, the model is trained from scratch for $\approx 8 \cdot 10^5$ steps. The initial learning rate is set to 10^{-4} and decreased by a factor of 10 after 400k steps. The Adam optimizer [KB14] is used as the optimizer, and a batch size of 1 is used for all runs. The model is trained on random crops of size 256×256.

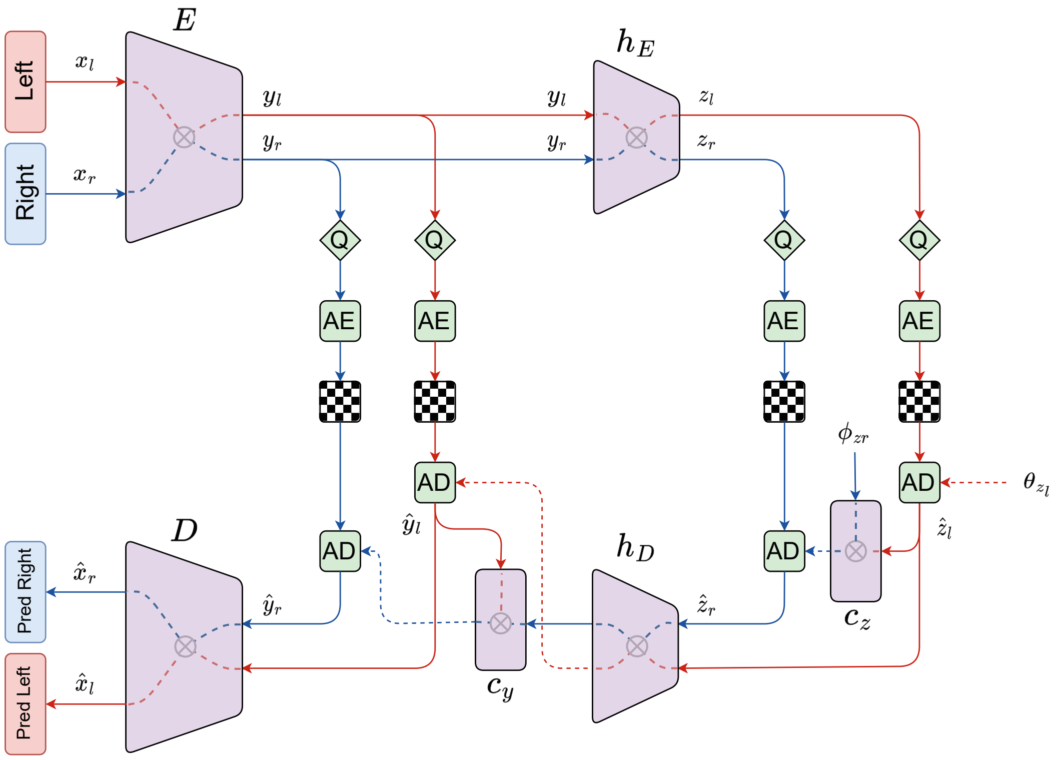


Figure 3.14: An overview of the architecture of the ECSIC model. The left and right streams are coloured red and blue, respectively. The encoder E, decoder D, hyperprior encoder h_E, and decoder h_D jointly process the left and right image streams and run in parallel. The stereo context modules c_y and c_z are only included in the right stream and use input from the left stream as contextual information. The Stereo Cross Attention (SCA) modules connect the left and right streams (shown as $\otimes$). Submodules in green (Quantizers (Q) and Arithmetic Encoder/Decoder (AE/AD)) do not contain trainable parameters. The bitstreams are marked with a checkerboard pattern. Dashed lines connecting to AD indicate predicted entropy parameters.

3.3.2.2 ECSIC

The proposed method follows the common structure [BMS+18] consisting of the main autoencoder and hyperprior, to which two non-autoregressive context modules are added; see overview in Figure 3.14. In the main branch, consisting of the encoder E and the decoder D, the input image pair $(\boldsymbol{x}_l, \boldsymbol{x}_r)$ is transformed into the latent representation $(\boldsymbol{y}_l, \boldsymbol{y}_r)$ and quantized to discrete tensors $(\hat{\boldsymbol{y}}_l, \hat{\boldsymbol{y}}_r)$, which form the bitstream. The decoder D reconstructs the output images $(\hat{\boldsymbol{x}}_l, \hat{\boldsymbol{x}}_r)$. In the hyperprior branch, the hyperencoder h_E transforms the latents into $(\boldsymbol{z}_l, \boldsymbol{z}_r)$, which are again quantified into $(\hat{\boldsymbol{z}}_l, \hat{\boldsymbol{z}}_r)$ and stored in the bitstream as side information. They are then used by the hyper-decoder h_D to

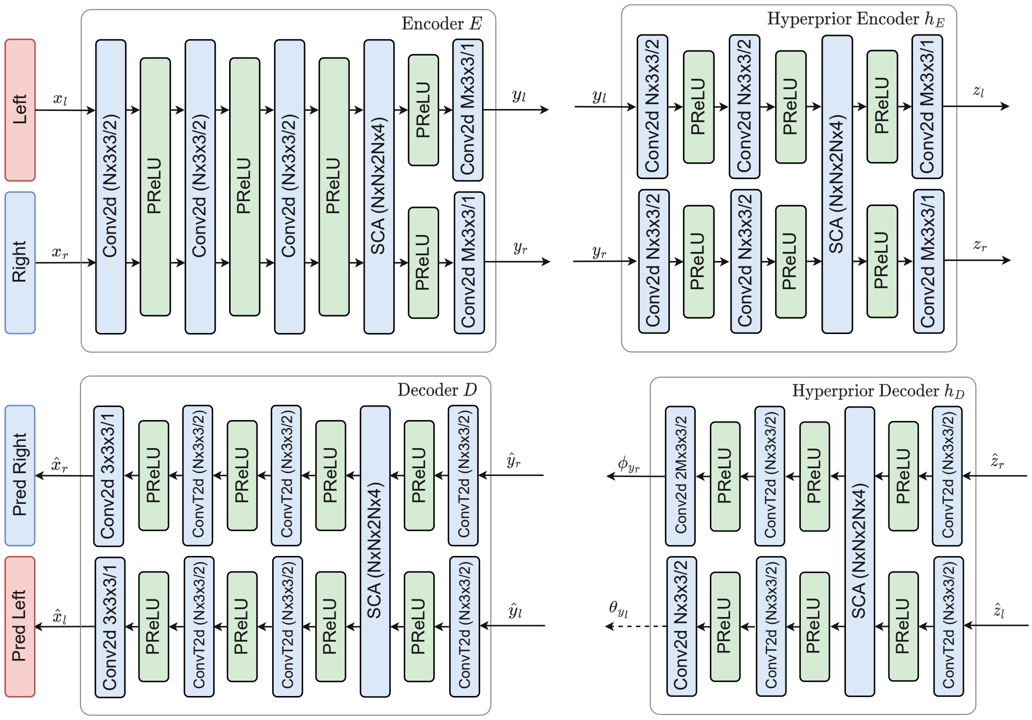


Figure 3.15: The left columns shows encoder E and decoder D. The right columns show the encoder and decoder of the hyperior h_E and h_D. $N = 192$ and $M = 48$ for all experiments. Conv2d denotes 2d convolutional layers, and ConvT2d 2d transposed convolutional layers. The initial three convolutional and PReLU layers in the encoder have shared weights between the left and right streams.

estimate the entropy parameters of the latents $(\hat{\boldsymbol{y}}_r, \hat{\boldsymbol{y}}_r)$. All these modules jointly process the left and right images in parallel.

Two non-autoregressive *stereo context modules* c_y and c_z are added to the right image stream to help estimate the entropy parameters of the right image latents $\hat{\boldsymbol{y}}_r$ and hyperlatents $\hat{\boldsymbol{z}}_r$ respectively, using the information already available from the left side. The left and right image streams are further connected by the proposed *Stereo Cross Attention* (SCA) modules (see Section 3.3.2.2), which are included in all modules that connect both streams – in encoder/decoder and hyper-encoder/hyper-decoder —and also in the stereo context c_y.

The resulting method can be trained end-to-end with the rate-distortion loss (see Sec. 3.3.2.2) on any dataset of stereo image pairs. Unlike other recent methods [DYY$^+$21, ZSZ23], the proposed method does not include any autoregressive components, which allows for fast encoding and decoding (see Sec. 3.3.3.3).

Encoding modules and quantization The encoder E and the decoder D each consist of four convolutional layers with three down/upsampling steps each for left and right, a SCA module, and PReLU [HZRS15] activation functions. The hyperprior encoder and decoder also use three convolutional layers with two down/upsampling steps each, an SCA module, and PReLU activation functions. The first convolutional layers in the encoder E have shared weights between the left and right streams. Network diagrams for each module can be found in Figure 3.15. Each quantization operation applies integer rounding to the mean-subtracted input. For example, for the left latent:

$$\hat{\boldsymbol{y}}_l = \mathrm{round}(\boldsymbol{y}_l - \boldsymbol{\mu}_l) + \boldsymbol{\mu}_l, \tag{3.18}$$

where $\boldsymbol{\mu}_l$ is the estimated mean of the distribution of $\boldsymbol{y}_l$. Analogously for $\boldsymbol{y}_r, \boldsymbol{z}_l$, and $\boldsymbol{z}_r$.

Stereo Cross Attention Module A new Stereo Cross Attention (SCA) module is proposed to facilitate the flow of non-local information between the left and right image compression streams. It performs cross-attention between the corresponding epipolar lines. For each location in one image, the attention domain is the corresponding horizontal row in the other image. By restricting attention to the corresponding epipolar lines (restricting it to horizontal lines for rectified images), the problem of quadratic memory complexity of vanilla attention can be circumvented, and all are processed in parallel. The resulting method still has quadratic complexity, but only in the width rather than the total number of pixels $\mathcal{O}(w^2h)$. The structure of the SCA module is shown in Figure 3.16. The layer norm is only applied to queries and keys (not values, which are the final output). In the Multi-Head Attention (MHA) block, 1D convolutions with a kernel size of 3 are used instead of linear embeddings. Other variants of position encoding [SUV18, VSP+17b] were explored, but no impact on overall performance was found. The SCA module is included in all submodules that combine both streams in E, D, h_E, h_D, and also in the stereo context c_y; see Figure 3.14. In E and h_E, the SCA module is applied after all downsampling layers and before the final convolutional layer. In D and h_D, the module is applied after the first upsampling layer.

Entropy estimation Following the hyperprior structure of [BMS+18], a pair of hyperlatents $\hat{\boldsymbol{z}}_l, \hat{\boldsymbol{z}}_r$ is used as side information for the entropy parameter estimation of the main latents $\hat{\boldsymbol{y}}_l, \hat{\boldsymbol{y}}_r$. In the following paragraphs, tensors (non-scalars) are written in bold, $\boldsymbol{\theta}_{(\cdot)}$ denotes entropy parameters and $\boldsymbol{\phi}_{(\cdot)}$ other learnable or predicted parameters.

The distribution of the left hyperlatent $\hat{\boldsymbol{z}}_l$ is modelled by a channelwise Laplacian distribution $\mathrm{Lap}_{\boldsymbol{\mu},\boldsymbol{b}}$ with parameters $\boldsymbol{\theta}_z^l := (\boldsymbol{\mu}_z^l, \boldsymbol{b}_z^l)$ for each channel of $\hat{\boldsymbol{z}}_l$ learned during training and fixed afterwards. The distribution of the right hyperlatent is modelled by a factorized Laplace distribution with parameters $\boldsymbol{\theta}_z^r := (\boldsymbol{\mu}_z^r, \boldsymbol{b}_z^r)$ for each pixel. These are predicted adaptively for each input. Similarly, the distribution of the main latents $\hat{\boldsymbol{y}}_{l/r}$ is also modelled by a factorised Laplace distribution with parameters $(\boldsymbol{\theta}_y^l, \boldsymbol{\theta}_y^r)$.

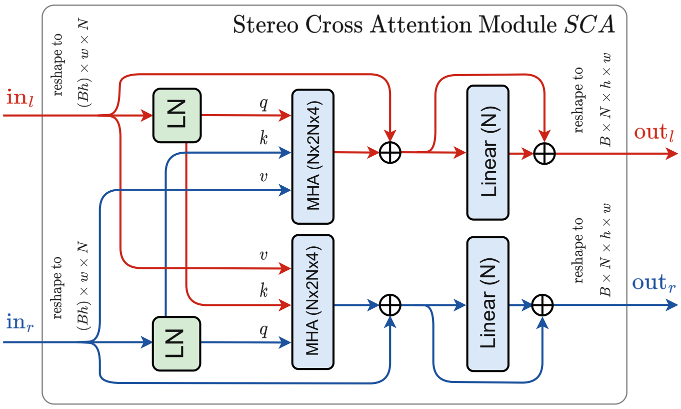

Stereo Cross Attention Module SCA
in$_l$
in$_r$
reshape to $(Bh)\times w\times N$
reshape to $(Bh)\times w\times N$
LN
LN
q
k
v
v
k
q
MHA (Nx2Nx4)
MHA (Nx2Nx4)
Linear (N)
Linear (N)
reshape to $B\times N\times h\times w$
reshape to $B\times N\times h\times w$
out$_l$
out$_r$

Figure 3.16: The proposed Stereo Cross Attention (SCA) module. The left and right streams are coloured red and blue, respectively. LN denotes layer norm, and MHA denotes multi-head attention block with arguments (output dimension × embedding dimension × heads). The streams denoted q, k, and v refer to the standard query, key and value terminology.

To reduce the bitrate, the right image entropy model is conditioned on information from the left stream. Two stereo context modules, c_y and c_z, are included for this purpose; see Figure 3.17.

The left hyperlatent entropy parameters are learned. The right hyperlatent entropy parameters $\boldsymbol{\theta}_z^r$ are predicted by c_z from $\hat{\boldsymbol{z}}_l$ and a set of fixed (learnable) parameters $\boldsymbol{\phi}_{\boldsymbol{z}_r}$:

$$\boldsymbol{\theta}_z^r = c_z(\hat{\boldsymbol{z}}_l, \boldsymbol{\phi}_{\boldsymbol{z}_r}). \tag{3.19}$$

During encoding (decoding), $\hat{\boldsymbol{z}}_l$ is first encoded (decoded) using its fixed entropy model and then used to encode (decode) $\hat{\boldsymbol{z}}_r$ using the entropy parameters predicted by c_z.

The parameters $\boldsymbol{\theta}_y^l$ of the distribution of the left latent $\hat{\boldsymbol{y}}_l$ are predicted from the two hyperlatents $\hat{\boldsymbol{z}}_l, \hat{\boldsymbol{z}}_r$ by the hyperprior decoder h_D. Similar to the previous case, the decoded left latent $\hat{\boldsymbol{y}}_l$ is used to aid entropy parameter estimation of the right latent. For this, the context module c_y is included, which predicts the entropy parameters of the right latent

$$\boldsymbol{\theta}_y^r = c_y(\hat{\boldsymbol{y}}_l, \boldsymbol{\phi}_{\boldsymbol{y}_r}) \tag{3.20}$$

from the already decoded left latent and the second output $\boldsymbol{\phi}_{\boldsymbol{y}_r}$ of the hyper decoder h_D.

Loss Function The rate-distortion loss is used

$$\mathcal{L} = \mathcal{R} + \lambda\mathcal{D}, \tag{3.21}$$

where $\mathcal{R}$ denotes the rate and $\mathcal{D}$ the distortion loss term; $\lambda \in \mathbb{R}$ is a trade-off parameter that determines the average bitrate of the trained model. The distortion loss term is the

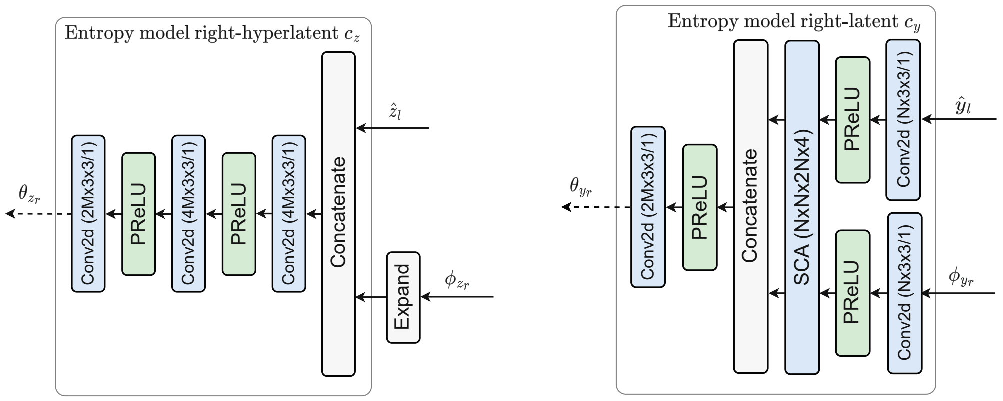


Figure 3.17: The context modules c_z and c_y. Top: c_z predicts the entropy parameters of the right hyperlatent $\hat{\boldsymbol{z}}_r$ from the left hyperlatent $\hat{\boldsymbol{z}}_l$ and a set of learned parameters $\boldsymbol{\phi}_{\boldsymbol{z}_r}$. Bottom: c_y predicts the entropy parameters of the right latent $\hat{\boldsymbol{y}}_r$ from the left latent $\hat{\boldsymbol{y}}_l$ and partial output of the hyper-decoder h_D denoted $\boldsymbol{\phi}_{\boldsymbol{y}_r}$. In the experiments, $N = 192$ and $M = 48$. The arguments after the convolutions denote (output dimension × kernel/stride).

expectation of the mean squared errors

$$\mathcal{D}(\boldsymbol{x}_l, \boldsymbol{x}_r) = \mathbb{E}_{\boldsymbol{x}_l, \boldsymbol{x}_r \sim p_{\boldsymbol{x}}} \Big[\|\boldsymbol{x}_l - \hat{\boldsymbol{x}}_l\|_2^2 + \|\boldsymbol{x}_r - \hat{\boldsymbol{x}}_r\|_2^2 \Big]. \tag{3.22}$$

The estimated rate is given by the cross-entropy between the predicted distribution of the entropy model and the true distribution of the latents/hyperlatents. The total rate loss is then the sum of the rates of the latents and the hyperlatents:

$$\begin{aligned} \mathcal{R} = \mathbb{E}_{\boldsymbol{x}_l, \boldsymbol{x}_r \sim p_{\boldsymbol{x}}} \big[& - \log_2 p(\hat{\boldsymbol{z}}_l \mid \boldsymbol{\theta}_z^l) \\ & - \log_2 p(\hat{\boldsymbol{z}}_r \mid \boldsymbol{\phi}_{c_z}, \boldsymbol{\phi}_{\boldsymbol{z}_r}, \hat{\boldsymbol{z}}_l) \\ & - \log_2 p(\hat{\boldsymbol{y}}_l \mid \boldsymbol{\phi}_{hd}, \hat{\boldsymbol{z}}_r, \hat{\boldsymbol{z}}_l) \\ & - \log_2 p(\hat{\boldsymbol{y}}_r \mid \boldsymbol{\phi}_{c_y}, \boldsymbol{\phi}_{hd}, \hat{\boldsymbol{y}}_l, \hat{\boldsymbol{z}}_r, \hat{\boldsymbol{z}}_l) \big], \end{aligned} \tag{3.23}$$

where $\boldsymbol{\phi}_{hd}, \boldsymbol{\phi}_{c_y}, \boldsymbol{\phi}_{c_z}$ denote the parameters of the hyperprior decoder and the proposed stereo context modules c_y and c_z respectively, and $p(\ldots)$ are the Laplace distributions specified in the previous Section 3.3.2.2.

Since the quantization operation has a zero derivative almost everywhere, it must be replaced by some proxy expression during training. As in [BLS17], the quantization operation is approximated with additive uniform noise for the rate loss (similarly for $\boldsymbol{y}_r, \boldsymbol{z}_l$ and $\boldsymbol{z}_r$).

$$\tilde{\boldsymbol{y}}_l = \boldsymbol{y}_l + \mathcal{U}(-0.5, 0.5). \tag{3.24}$$

Following Minnen et al. [MS20], a straight-through estimation quantization is used for the distortion loss during training.

Training The ECSIC model is trained for $\approx 1.3 \cdot 10^6$ steps. The Adam optimizer is used, and the initial learning rate is set to 10^{-4} for batches of size 1 and reduced to 10^{-5} after 10^6 steps. The λ values are adjusted according to the target bitrate.

3.3.3 Evaluation

This section begins with a summary of the data sets used to evaluate the proposed methods, followed by details on the training parameters. This is followed by a presentation of the rate-distortion curves and an ablation study.

Datasets: The proposed methods are evaluated using two popular stereo image datasets, namely Cityscapes [COR+16] and InStereo2k [BWX+20], which each provide different stereo image environments. Cityscapes consists of different disparity stereo pairs from driving scenarios, while InStereo2k is dedicated to indoor environments with objects positioned closer to the camera. The Cityscapes dataset, comprising 5000 urban street scene stereo pairs taken in German cities, provides images with a resolution of 2048×1024, divided into 2975 image pairs for training, 500 for validation and 1525 for testing. The images are cropped by 64, 256 and 128 pixels from the top, bottom and sides, respectively, to remove vehicle components and rectification artefacts. The InStereo2k dataset, which contains 2060 stereo images of indoor scenes, divides its 1080×860 resolution images into 2010 for training and 50 for testing. These images are symmetrically cropped in a minimal way to ensure that height and width are multiples of 32.

Codecs: The methods are evaluated against a comprehensive collection of both traditional and learned codecs, which can be categorized into single-image compression (BPG), video compression (HEVC), multi-view compression (MV-HEVC, LDMIC) and stereo image compression (HESIC/HESIC+, DSIC). For BPG [Bel14], each frame is processed separately, and chroma subsampling is omitted. The video codecs HEVC [SOHW12] and VVC [BWY+21] process stereo image pairs as two-frame video sequences, again with chroma subsampling disabled to prevent PSNR degradation. For HEVC, the reference implementation[5] is used with the `main_444_12` profile. VVC results are from Zhang et al. [ZSZ23], using the `lowdelay_p` setup and YUV444 format. MV-HEVC[6] [MMSW06] is used in its two-view intra-mode configuration and only supports 4:2:0 chroma mode, which affects PSNR at higher bitrates. Results for learned methods such as DSIC [LWU19a] and HESIC+ [DYY+21] are derived from their respective publications. The LDMIC [ZSZ23] results are shown for both the full LDMIC model, including an autoregressive context model, and the simplified LDMIC (fast) without the autoregressive elements.

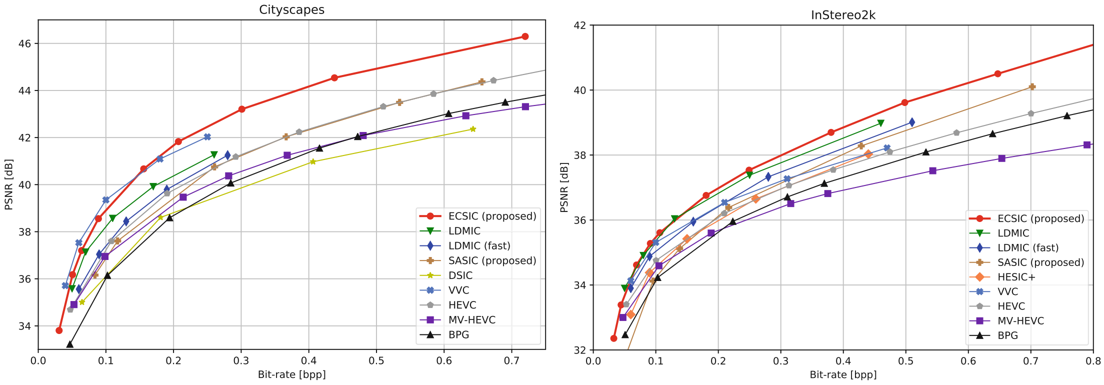


Figure 3.18: Rate-distortion curves of SASIC and ECSIC compared to other codecs on Cityscapes (left) and InStereo2K (right) datasets measured by PSNR and MS-SSIM.

Table 3.6: Relative quality difference (PSNR gain at the same bitrate; higher is better) and bitrate difference (bitrate gain for the same PSNR; lower is better) of the benchmarked methods w.r.t. BPG. The best results are in bold, and the second best are underlined.

	Cityscapes		InStereo2k	
Method	BD-PSNR [dB]↑	BD-Rate [%]↓	BD-PSNR [dB]↑	BD-Rate [%]↓
ECSIC	2.86	-51.90	**1.57**	**-42.08**
LDMIC	2.07	-42.20	1.26	-41.03
LDMIC (fast)	1.35	-29.66	0.87	-30.40
SASIC	0.98	-22.40	0.38	-15.43
DSIC	0.07	-3.35	-	-
HESIC+	-	-	0.37	-14.90
VVC	**3.12**	**-56.24**	0.86	-31.02
HEVC	1.14	-25.78	0.45	-15.09
MV-HEVC	0.41	-10.07	0.19	-4.96
BPG	0.0	-0.0	0.0	-0.0

3.3.3.1 Rate-Distortion Curves

The Peak Signal to Noise Ratio (PSNR) rate-distortion curves can be seen in Figure 3.18. In addition, Bjøntegaard Delta bitrate (BD-rate) and BD-PSNR scores [Bjo01a] are reported for each codec vs. BPG for Cityscapes and InStereo2K in Table 3.6. On InStereo2k, ECSIC outperforms all other codecs tested. On Cityscapes, ECSIC performs worse than VVC for low bpp (bits per pixel) (< 0.15) but is the first learned method to outperform VVC on PSNR for bpp > 0.15. All other codecs are outperformed by ECSIC. The second best learned method, LDMIC, relies on autoregressive entropy modelling in the default version, rendering it much slower than SASIC and ECSIC (see Section 3.3.3.3). The fast version of LDMIC without autoregressive context performs much worse with a BD-rate compared to ECSIC of 24.35% for InStereo2k and 49.23% for Cityscapes. At the time of their respective releases, SASIC and ECSIC were the best-performing learned methods for Cityscapes and InStereo2k while performing on par with HEVC on Cityscapes.

ECSIC also outperforms the other learned SIC models, DSIC and HESIC+. HESIC+ relies on explicit warping to remove spatial redundancy by using a homography to warp between the left and right images. While both SASIC and ECSIC use a cross-attention connection in the decoder, the ECSIC model does not use explicit warping and additional cross-attention connections during encoding and in the entropy modelling submodules. The performance gap between SASIC and ECSIC shows the effectiveness of the additional modules and the new SCA layer. See Appendix B for qualitative results on a selection of the Cityscapes and InStereo2k test images. ECSIC also outperforms the other learned SIC models, DSIC and HESIC+. HESIC+ relies on explicit warping to remove spatial redundancy by using a homography to warp between the left and right images. While both SASIC and ECSIC use a cross-attention connection in the decoder, the ECSIC model does not use explicit warping and additional cross-attention connections during encoding and in the entropy modelling submodules. The performance gap between SASIC and ECSIC shows the effectiveness of the additional modules and the new SCA layer. See Appendix B for qualitative results on a selection of the Cityscapes and InStereo2k test images.

3.3.3.2 Ablatation Study

This section provides an ablation study that examines the effects of the submodules of SASIC in Section 3.3.3.2 and of ECSIC in Section 3.3.3.2. In Section 3.3.3.3, the encoding and decoding runtimes are evaluated and compared with other methods.

SASIC This section compares the submodules of the SASIC model and examines their effect on the overall rate-distortion curves. A comparison of these cases can be seen in Figure 3.19. In addition, Bjøntegaard Delta PSNR (BD-PSNR)[Bjo01b] and BD-Rate

[5] https://vcgit.hhi.fraunhofer.de/jvet/HM
[6] http://hevc.info/mvhevc

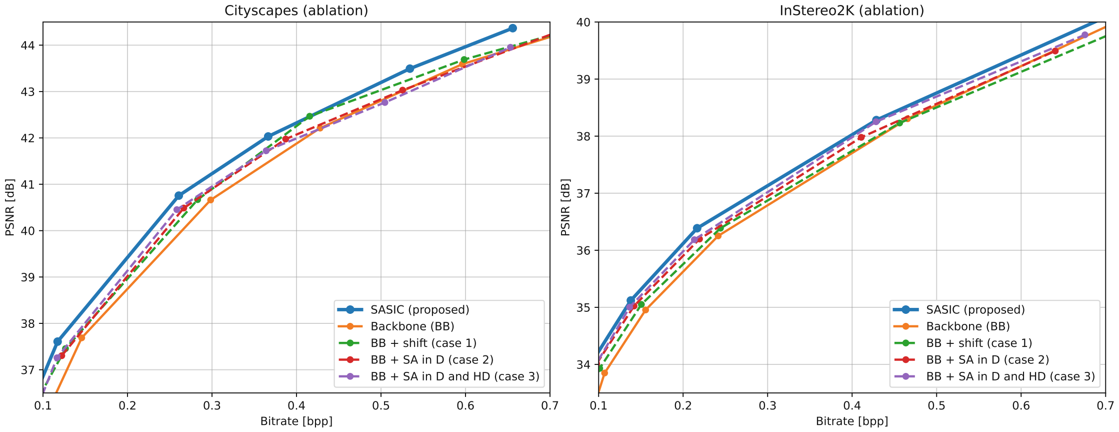


Figure 3.19: Ablation study: Comparison of the effects of the latent shift residual coding (green) and the stereo attention sub-modules (SA) on the rate-distortion performance. SA is used only in the image decoder (red) or the hyperprior decoder (purple). The full proposed method (blue) and the original backbone (orange) are shown for reference.

Table 3.7: Comparison of BD-Rate (lower is better) and BD-PSNR (higher is better) between the backbone model and each of the cases.

	Cityscapes		InStereo2k	
Method	BD-PSNR [dB]↑	BD-Rate [%]↓	BD-PSNR [dB]↑	BD-Rate [%]↓
SASIC	-23.42	1.05	-11.28	0.38
Case 1	-14.58	0.67	-2.28	0.07
Case 2	-19.70	0.80	-10.6	0.31
Case 3	-17.78	0.73	-8.97	0.28
Backbone	0.0	0.0	0.0	0.0

values are given in Table 3.7. BD-PSNR approximates the quality gain for equivalent bitrate (higher is better), and BD-Rate approximates the bitrate saving percentage for equivalent quality (negative and lower is better).

- **SASIC (proposed):** The SASIC model combines all the improvements, i.e. backbone + shift + stereo attention in the decoder and hyperprior decoder. It is identical to the SASIC model in Figure 3.18.

- **Backbone:** In the Backbone model, both images are compressed independently, with the model used to compress the left image in the SASIC model.

- **Backbone + shift (case 1):** In this case, the stereo attention modules connecting D_1 and D_2 in Figure 3.11 are removed, leaving only the connections between $\hat{\boldsymbol{y}}_1$ and $\boldsymbol{y}_2$ and h_D^{res}. After training, it can be seen from the RD-curves in Figure 3.19 that the model performs significantly worse than the full model, indicating that the stereo attention in the decoder helps to reduce the bitrate further. The BD rate in table 3.7 shows that the remaining connection still gives a significant improvement for Cityscapes compared to the backbone model where no such links are present. For InStereo2k, the improvements are smaller.

- **Backbone + stereo attention in decoder (case 2):** In this case the connection between $\hat{\boldsymbol{y}}_1$ and $\boldsymbol{y}_2$ as well as h_D^{res} is removed. The decoder of the hyperprior model for the right image is replaced by the decoder of the left model (Figure 3.12 bottom left) while keeping the stereo attention connection between D_1 and D_2. The resulting model performs significantly better than the backbone but still worse than the full SASIC model.

- **Backbone + stereo attention in decoder and hyperprior decoder (case 3):** To demonstrate the effectiveness of the connections in the latent, which are not present in *case 2*, an architecture based only on a stereo attention connection is investigated in this case. Three connections are added between the hyperprior decoders in *case 2*, similar to the decoder connections. From Table 3.7, it can be seen that stereo attention alone is inferior in performance to the full SASIC model. The resulting model performs even worse than the simpler *case 2* model.

ECSIC This paragraph compares the submodules of the ECSIC model and examines their effects on the overall rate-distortion curves. A comparison of these cases can be seen in Figure 3.20. In addition, Bjøntegaard Delta PSNR (BD-PSNR)[Bjo01b] and BD-Rate values are given in Table 3.8 for Cityscapes and in Table 3.9 for InStereo2k.

- **Backbone:** To assess the impact of individual components on stereo compression performance, each ablation of the ECSIC model is compared to a *Backbone* model obtained by stripping the ECSIC model of both context modules c_y and c_z (the entropy modelling for left and right is independent of each other) and removing each SCA module in the remaining architecture (including the corresponding activation functions of each SCA layer). The result is two separate models that compress the left and right images of a stereo image pair independently.

- **ECSIC (proposed):** The proposed method with all additions as shown in Figure 3.14. The biggest gains over the backbone model are at low bit rates. For example, for Cityscapes, the BD-rate limited to low PSNR ($34 - 38$dB) shows a bitrate saving of 37.0%, while in the high PSNR range ($44 - 46$dB), the difference is reduced to 19.0%. The maximum asymptotic theoretical bitrate saving that can be achieved is 50.0%, which corresponds to compressing a stereo pair at the bitrate of a single frame. In reality, the optimum is even lower due to occlusions and non-overlapping fields of view in the stereo pair.

- **Only encoder SCA:** The backbone model is extended by adding a single SCA layer to the encoder E. The resulting model shows no significant improvement over

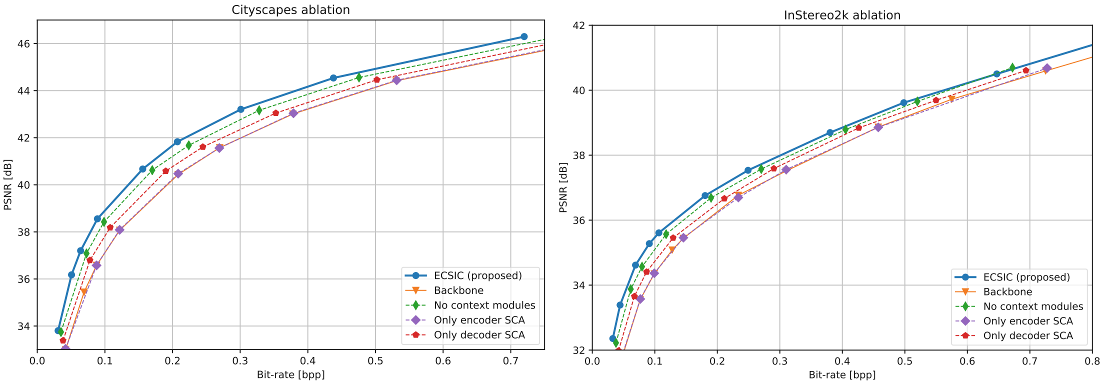


Figure 3.20: Rate-distortion curves for ECSIC with varying modifications for Cityscapes (left column) and InStereo2k (right column) measured by PSNR.

the backbone. However, it has been found that adding SCA modules in the encoder improves performance if corresponding SCA modules are present in the decoders.

- **Only decoder SCA:** The backbone model is extended by adding a single SCA layer in the decoder D. Contrary to only adding SCA in the encoder, adding a single SCA layer in the decoder already gives an improvement of 11.7% over the backbone method on Cityscapes.
- **No context modules:** Removing the context modules c_y and c_z results in a rate reduction of 12.5% compared to the full ECSIC model.

The comparison shows that both the proposed context modules c_y and c_z, as well as the proposed SCA modules, enable better compression of stereo images when compared to the single image compression backbone model that compresses both images independently. Furthermore, the experiments suggest that the SCA module works best in the decoding parts of the model. However, it was found that SCA modules in the encoding part of the model, in conjunction with SCA modules in the corresponding decoding parts, lead to the best overall performance. Experiments were also conducted with different variants of positional encoding [SUV18, VSP$^+$17b], but no impact on compression performance was observed.

3.3.3.3 Runtimes

Fig. 3.21 shows the average encoding and decoding times of SASIC and ECSIC compared to other methods on the InStereo2k dataset (i.e. the images are already rectified). The conventional methods BPG, HEVC and MV-HEVC were evaluated on a single-core Intel Xeon Gold 6230R processor (times taken from Zhang et al. times [ZSZ23]). For LDMIC, their reported encoding and decoding times [ZSZ23] (measured on an NVIDIA RTX 3090 GP U) are shown. For SASIC and ECSIC, the encoding and decoding times are

Table 3.8: Relative quality difference (PSNR gain at the same bitrate; higher is better) and bitrate difference (bitrate gain for the same PSNR; lower is better) of the benchmarked methods on Cityscapes with respect to the backbone model. BD-Rates restricted to a low PSNR range (34 – 38dB) and a high PSNR range (44 – 46dB) are also reported.

Method	BD-PSNR [dB]↑	BD-Rate [%]↓	BD-Rate [%]↓ low PSNR
ECSIC (proposed)	**1.49**	**-30.18**	**-36.96**
No context modules	1.02	-21.45	-26.95
Only decoder SCA	0.54	-11.72	-15.57
Only encoder SCA	0.02	-0.40	-0.36
Backbone	0.0	-0.0	-0.0

Table 3.9: Relative quality difference (PSNR gain at the same bitrate; higher is better) and bitrate difference (bitrate gain for the same PSNR; lower is better) of the benchmarked methods on InStereo2k with respect to the backbone model. BD-Rates restricted to a low PSNR range (32 – 36dB) and high PSNR range (38 – 40dB) are also reported.

Method	BD-PSNR [dB]↑	BD-Rate [%]↓	BD-Rate [%]↓ low PSNR
ECSIC (proposed)	**0.77**	**-19.96**	**-37.04**
No context modules	0.63	-18.20	-27.67
Only decoder SCA	0.32	-9.36	-15.57
Only encoder SCA	0.0	-0.70	-2.53
Backbone	0.0	-0.0	-0.0

measured on an NVIDIA RTX 3090 GPU. The proposed method shows low encoding and decoding times, beating all other methods in this benchmark.

3.3.4 Conclusion

In this section, two methods for stereo image compression have been presented.

The SASIC model adapts a single image compression model with a hyper-prior entropy model with two additions: 1) a global shift and warp in the latent domain so that only the residual for the right image is encoded and 2) stereo attention connections in the decoder. Both modifications have been shown to improve compression performance in the ablation study. The resulting model achieved SOTA on Cityscapes and InStereo2k at the time of publication.

Similar to the SASIC model, the ECSIC model can be considered an adaptation of a single image compression model with a hyperprior entropy model. Unlike SASIC, it does not include explicit warping but instead makes extensive use of stereo attention

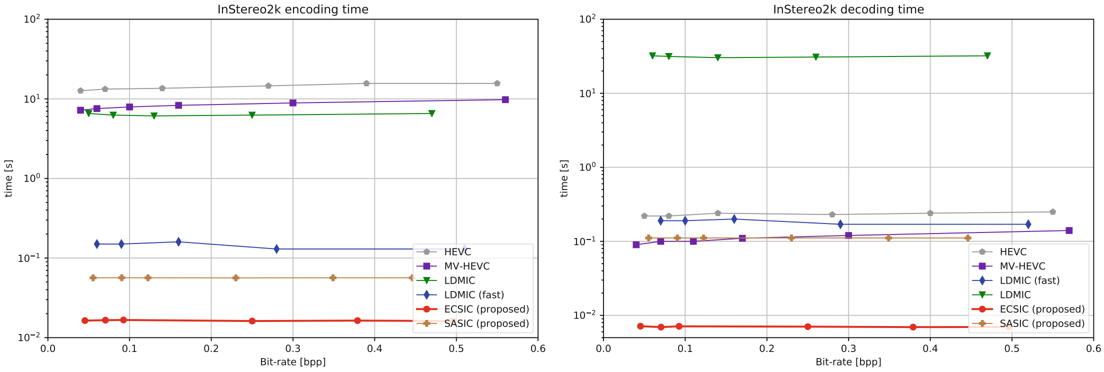


Figure 3.21: Encoding and decoding times of SASIC and ECSIC against other codecs on a logarithmic scale. The reported times are averages over the InStereo2k test set. All learned methods run on a GPU.

connections to allow implicit modelling of any disparity warping. In addition, two stereo context modules were proposed to improve entropy modelling. The effectiveness of these components was demonstrated in the ablation study. The resulting model is fast in encoding and decoding and outperforms all other learned compression methods on the two benchmark stereo image datasets, Cityscapes and InStereo2k.

Modelling redundancies between the left and right images in a stereo image pair for image compression is challenging for several reasons. Firstly, using disparity warping to reduce redundancy requires transmitting disparity maps over the bitstream, increasing the bitrate, and any inaccuracies in the predicted disparity maps introduce additional noise into the model, further increasing the required bitrate. However, less powerful warping approaches such as homographies[DYY^{+}21], while requiring significantly fewer bits to transmit (in fact, they are practically negligible for all but very low bitrates), are not powerful enough. Architecturally, models based solely on CNNs lack data specificity and require computationally expensive operations like 3D convolutions[LWU19b]. Stereo attention provides a solution to these problems, allowing for learned accurate modelling of stereo disparities without the need to transfer additional metainformation. However, the naive application of the attention operation is computationally infeasible for all but small images due to the quadratic memory and runtime complexity of the attention operation. A solution can be found by restricting the attention to the epipolar line, reducing the complexity from quadratic in pixel count to linear in width. For rectified stereo image pairs, where matching points between left and right images in a stereo image pair can always be assumed to lie on the same epipolar line, this is equivalent to restricting the attention operation to horizontal lines. This allows a significant simplification of the stereo cross-attention operation. Such a cross-attention operation can be useful for encoder and decoder modules and entropy modelling, as seen in the ablation studies in this section.

3.4 Summary

This chapter comprised three different applications, each of which demonstrates a different form of long-range dependency modelling using some form of attention.

Section 3.1 investigates the task of detecting textual baselines in historical document images. Textual baselines are linear sequences on document images that can span over the entire page. The proposed method is inspired by works on visual attention [XBK+15, JSZ+15] and extracts text baselines using a recurrent architecture from a sequence of extracted image patches by following the baseline.

Section 3.2 tackles the task of blast cell detection in bone marrow flow cytometry samples from ALL patients. Existing methods process samples event-wise, i.e. every cell is classified independently, resulting in fixed decision boundaries. However, the underlying biological decision boundaries are not fixed, with different phenotypes and even different flow cytometry machines or operators having a significant impact on the distribution of blast cells in a sample [RRK+16]. The Flowformer model is a variant of a Transformer than can process entire samples at once. It can, therefore, capture relative dependencies between different events in a sample and global structures. Table 3.5 shows that it is also capable to generalize to data from different clinics and operators to some extent, while being trained on only 66 samples.

Finally, in Section 3.3, two methods for stereo image compression are presented. Stereo image pairs typically have overlapping fields of view, resulting in a high amount of mutual information that allows potentially more efficient compression than compressing both images separately. However, this requires accurate matching of corresponding objects and points between the two images. The SASIC and ECSIC models use attention to model the dependencies between the left and right images without the need for separate disparity estimation. The resulting methods outperform other methods on two public stereo image datasets [COR+16, BWX+20].

CHAPTER 4

Conclusion

Three different applications have been investigated in this thesis, each of which requires accurate modelling of a different set of long-range dependencies.

The first application, modelling text baselines in handwritten documents, requires the modelling of long linear sequences in the document images. This thesis proposes a model that uses visual attention to follow the sequence through the image. Given the starting points of the text lines, this method allows end-to-end processing of the document image, avoiding the need for heuristics to extract baseline coordinates. Unlike other methods at the time, this allows direct prediction of baseline coordinates without relying on heuristics to extract the coordinates from a segmentation output [GLS^{+}19]. The proposed model processes document images in sequences of local images, keeping memory requirements low. However, the starting points of the baselines are required to initiate the recurrent process. As shown in Table 3.3, the model can be used without any segmentation and as a follow-up to any baseline detection method. However, not using the segmentation results leads to a small drop in detection performance. A disadvantage of this method is that it requires start points of text baselines for the recurrent model to work, and relying on a separate segmentation model requires a separate training step and additional memory requirements. Follow-up work could improve this by extracting potential start points using a region proposal network instead of a more expensive segmentation model.

The second application, modelling cell populations in flow cytometry data, requires modelling global dependencies in high dimensional point clouds. Associating these high-dimensional point clouds with a "bag of word embeddings" motivated the model proposed in Chapter 3.2, where a Transformer model was introduced that takes cell measurements directly as input without any positional embedding, allowing flow cytometry samples to be processed in a single forward pass. However, these samples often contained up to 1M cells, which made a naive application of Transformers difficult due to the quadratic growth in complexity. The model presented in this thesis, therefore, makes use of a variant of the Transformer layer initially proposed by Lee et al. [LLK^{+}19a], which introduces a fixed length, learned "intermediate set" of k induced points, resulting in a reduction in complexity from $\mathcal{O}(n^2)$ to $\mathcal{O}(nk)$. The resulting model successfully generalizes to data from other hospitals, demonstrating that the model accurately models global structures rather than relying solely on position to classify cells like existing neural models [LSR+18]. The model called *Flowformer* was the first neural network to achieve SoTA results on this task. The model is lightweight, with only $\approx 28k$ parameters, allowing it to run on a CPU, even on older hospital hardware. The main bottleneck for improving this method is the lack of training data. This is particularly true for the more challenging acute myeloid leukaemia, which is rarer than acute lymphoblastic leukaemia, resulting in significantly less training data while also being more heterogeneous. In addition, because the Transformer model is trained on a specific set of features (corresponding to cell surface markers), applying it to data from a laboratory with different markers requires retraining or ignoring unseen features. Therefore, follow-up work is mainly concerned with removing this requirement, which would improve the generalizability of the method while allowing training on more diverse training sets [WKR+24]. Another line of research not explored in this thesis is general pretraining on large amounts of unlabeled data. Online collections such as the *Flowrepository*[1] contain large quantities of flow cytometry from different laboratories around the world. Since Transformer models, in particular, have been shown to exhibit impressive scaling behaviour as the amount of training data [KMH+20] increases, extensive pretraining could help the model to generalize. However, unlike NLP data, which contains sequential data where autoregressive pretraining is possible, flow cytometry data lacks this linear structure, which requires a different pretraining objective. These future directions are mainly concerned with improving the performance of these methods on unseen data. Another potential research direction not discussed in this thesis is the explainability of these methods. The transformer model presented in this thesis does not provide an explanation for its prediction and can be considered a black box from a clinician's perspective, making it impossible to rely entirely on its predictions in clinical routine. Follow-up work by our group has led to a modification of the Transformer model in this thesis that, instead of modelling blast cell prediction as a binary classification problem, reproduces the manual *gating* process used in clinical routine [KWW+22]. This is achieved by replacing the linear classification decoder layer in the model from Section 3.2 with a DETR [CMS+20] inspired decoder block that directly predicts a sequence of polygons that iteratively restricts the data space.

The third application, the modelling of stereo redundancies for stereo image compression, requires the modelling of dependencies between a pair of stereo images. These dependencies are sample-dependent and can be highly non-local in high disparity regions. Furthermore, image compression models often require fast decoding speeds, mainly when used as part of a video compression codec, where real-time decoding is needed. This makes it challenging to model long-range dependencies with deep convolutional networks or networks based on computationally expensive 3D convolutions. The two methods presented in this thesis circumvent these problems by connecting the left and

[1] https://flowrepository.org/

right images in a stereo-image pair via cross-attention. By restricting attention to the epipolar line, the resulting layer allows efficient modelling of redundancies in stereo images. In addition, entropy models are proposed that allow conditioning of the right image on the left via one-way cross attention during entropy encoding and decoding. The experiments assume rectified images since the most popular public benchmark datasets such as Cityscapes [COR+16] and InStereo2k [BWX+20] consist only of rectified images, making the method conceptually simpler. In the case of an unrectified stereo image pair, the method would have to be adapted by computing the epipolar line on the fly and restricting attention to the epipolar line by sampling positions for key/value pairs from points on the corresponding line in the other image. Both methods achieved SoTA at the time of publication, both in terms of compression performance and speed (assuming a GPU; otherwise, traditional methods are still competitive for decoding speed). Unlike other learned stereo image compression models such as HESIC or DSIC, the proposed models are lightweight, allowing their use in resource-constrained environments. Unlike the recent LDMIC model, the proposed methods do not rely on autoregressive components, enabling fast encoding/decoding times. However, to achieve real-time coding, the proposed methods require a GPU, which may not be available depending on the application. While real-time is often not a strict requirement for image compression methods, real-time decoding speeds are required for video compression methods. Achieving real-time decoding is still an active area of research for learned compression methods, with the first successful real-time codecs for mobile devices being proposed only very recently [vRSL+24], although so far only for single image/video compression, and not for stereo image data, making an adaption of these ideas to stereo image data an attractive direction for future research. Another research direction not explored in this thesis is the compression of stereo video data. In principle, the proposed methods can be used as an intra-frame compression method of a video codec that treats left and right images separately for the remaining steps. However, since left and right images show mostly the same objects due to overlapping fields of view, the motion compensation vectors are expected to be highly similar, allowing further reduction in bitrate. Finally, the methods in this thesis are optimized for PSNR and MS-SSIM rather than for maximum perceptual quality. Measuring perceptual quality is an active area of research and a recurring task in the Challenge on Learned Image Compression (CLIC), the popular compression workshop held annually at CVPR. However, at the time of writing, no single approach has been established. Instead, methods optimizing for perceptual quality such as HiFi [MTTA20] typically use a linear combination of conventional metrics such as MSE and perceptual metrics such as Learned Perceptual Image Patch Similarity (LPIPS) [ZIE+18], VGG loss terms [SZ14] and a GAN [GPAM+20] loss term. The total loss is then a weighted mean of all these, where the weights are determined heuristically. Optimizing for perceptual metrics makes comparison with other methods difficult, and it is unclear what effect this has on potential downstream tasks such as follow-up detection or segmentation networks on the compressed stereo images. I believe finding a reliable perceptual metric would have the most significant potential impact on the future development of stereo-image compression and image and video compression in general.

The first application discussed, text baseline detection in handwritten documents, is based on recurrent visual attention inspired by early work by Xu et al. [XBK+15], and the rest of this thesis uses an attention mechanism inspired by the attention operation in the Transformer model [VSP+17a]. While visual attention allows for more memory-efficient processing of the input by only scanning the input in a sequence of smaller windows, the self-attention operation is notorious for its memory requirements, which grow quadratically with the input size. In return, self-attention (as well as most of its variants [TDBM20, TDA+20]), can be more naturally parallelized during training due to the absence of recurrent components, can be more naturally parallelized during training, allowing for better scaling and explaining the recent dominance of Transformers among foundation models for NLP [BMR+20b, AAA+23, VSP+17a]. For applications such as text baseline detection, where smaller architectures are sufficient, parallelizability is less critical than for large foundation models. One advantage of attention-based networks over convolutional networks in vision is that the former is data-specific [PK22]. Park et al.[PK22] identify data specificity as the primary reason for the generalization capabilities of attention-based architectures. Data specificity helps in modelling flow cytometry patterns by allowing the network to move beyond fixed decision boundaries and modelling stereo image redundancies by enabling the network to perform comparison operations to deal with input-dependent disparity distributions. However, as seen in this work, applications of self-attention networks often require additional care in handling the memory requirements of the attention operation. This can be done by working with modifications of the full attention operation, such as the *IAB* of Chapter 3.2 (there are now a number of variants, each with different trade-offs, see Tay et al. [TDBM20] for an overview), or by using application-specific prior knowledge to constrain the operation, such as the stereo attention module of Chapter 3.3, which used the fact that relevant points in the other image must necessarily be on the epipolar line.

Transformers have dominated much of natural language processing research since the seminal work of Vaswani et al. [VSP+17a]. While initially designed for text, recent work has introduced the Transformer model to vision, with ViT being its most prominent representative [DBK+20]. The scalability [HBM+22, KMH+20] and novel methods for pre-training [RKH+21, BMR+20a] have made attention-based models the architecture of choice for a wide range of tasks. However, scaling these models requires a trade-off between long-range dependencies and model size due to the quadratic memory requirements of self-attention, making the question of how to optimally model long-range dependencies one of the most pressing challenges of current foundational models, particularly for language. Therefore, the importance of efficiently modelling long-range dependencies extends beyond the specific applications discussed in this thesis, and recent developments in instruction-tuned language models have made the problem highly relevant to both research and industrial applications. As a result, many exciting ideas have emerged in the last two years that will likely form the basis of future research in this area. Notable examples include hardware-aware implementations of the Transformer such as FlashAttention [DFE+22, Dao23], which make efficient use of the memory hierarchy of some GPUs to achieve significant speedups without any loss of performance or RoPE scaling [LYZ+23, SAL+24], a method for extending the range of Transformers after pretraining.

In conclusion, this thesis contributes to the growing body of work on attention-based models and highlights their versatility and adaptability in three different applications. The thesis presents the challenges in overcoming them and hopefully helps the reader push the boundaries of what is possible with attention-based approaches.

APPENDIX A

Additional Results for Flowformer++

This chapter contains additional experimental results comparing the Flowformer and Flowformer++ models and detailed calibration results for Flowformer++.

A.1 Ablation study: Flowformer vs Flowformer++

This section contains additional training runs for the Flowformer model to allow a better comparison with the Flowformer++ model (which has different training configurations from the original model). Table A.1 shows the median F1 scores for the original Flowformer model (ep100), the model trained for 180 epochs (180ep), and an additional run (++training) where all training parameters match those of the Flowformer++ training. The performance drops compared to the original training runs. This is due to overfitting on the training sets, which was not observed for the Flowformer++ model. Additionally, on a qualitative note, the Flowformer++ training appears much more stable. The ablation study demonstrates that the better performance of the Flowformer++ model is not due to the longer training.

Table A.1: The median F_1-scores for each configuration. Boldface values indicate the best-performing method for a specific train/test dataset combination. The Flowformer 100ep is identical to the Flowformer method in Section 3.2.

train	**test**	**Flowformer**			**Flowformer++**	[RDS+19]
		100ep	180ep	++training		
vie	vie	0.94	0.89	0.91	**0.96**	-
bln	bue	**0.87**	0.67	0.80	0.83	0.68
	vie14	**0.90**	0.81	0.84	0.89	0.35
	vie20	0.87	0.75	0.81	**0.89**	0.48
bue	bln	0.77	0.71	0.78	**0.90**	0.50
	vie14	0.90	0.85	0.86	**0.93**	0.84
	vie20	0.88	0.75	0.85	**0.93**	0.86
vie14	bln	0.90	0.85	0.85	**0.94**	0.81
	bue	0.95	0.91	0.90	**0.98**	0.84
	vie20	0.89	0.83	0.85	**0.94**	0.86
vie20	bln	0.81	0.72	0.73	**0.95**	0.25
	bue	0.86	0.86	0.85	**0.96**	0.81
	vie14	**0.95**	0.91	0.91	**0.95**	0.89

A.2 Calibration

This section provides MRD curves and calibration histograms for the Flowformer++ model for the experiments discussed in Tab. 3.5. Tab. A.2 shows the Expected Calibration Error (ECE) for the Flowformer++ model. Formally, the Expected Calibration Error (ECE) is defined as follows: Let $B_1, B_2, \ldots, B_M$ be M equally-sized bins for the predicted probabilities, where in our case M is set to 10. For each bin B_m, we define:

- The number of predictions falling into bin B_m n_m.
- The total number of predictions N.
- The accuracy of the predictions in bin B_m $\text{acc}(B_m)$.
- The average confidence (predicted probability) of the predictions in bin B_m $\text{conf}(B_m)$.

The ECE is then calculated as:

$$\text{ECE} = \sum_{m=1}^{M} \frac{n_m}{N} |\text{acc}(B_m) - \text{conf}(B_m)| \tag{A.1}$$

Table A.2: Expected Calibration Error of the Flowformer++ model for the experiments discussed in Chapter 3.2.

train	**test**	**ECE**
vie	vie	0.051
bln	bue	0.141
	vie14	0.228
	vie20	0.068
bue	bln	0.181
	vie14	0.057
	vie20	0.221
vie14	bln	0.049
	bue	0.220
	vie20	0.250
vie20	bln	0.091
	bue	0.163
	vie14	0.060

This formulation captures the difference between the predicted probabilities (confidence) and the true accuracy across different probability ranges. A lower ECE indicates better calibration, with an ECE of 0 representing perfect calibration.

To compute the ECE, the predicted probabilities for all cells in the test set are collected in the bins $\{[0, 0.1], (1, 0.2], \ldots, (0.9, 1.0]\}$. Then, the mean of the absolute differences between the measured and expected probabilities is computed. For example, the expected probability for the bin $(0.2, 0.3]$ would be 0.25, meaning the probability that the cells predicted to belong to the positive class with probability $0.2 - 0.3$ should belong to the positive class on average with probability 0.25.

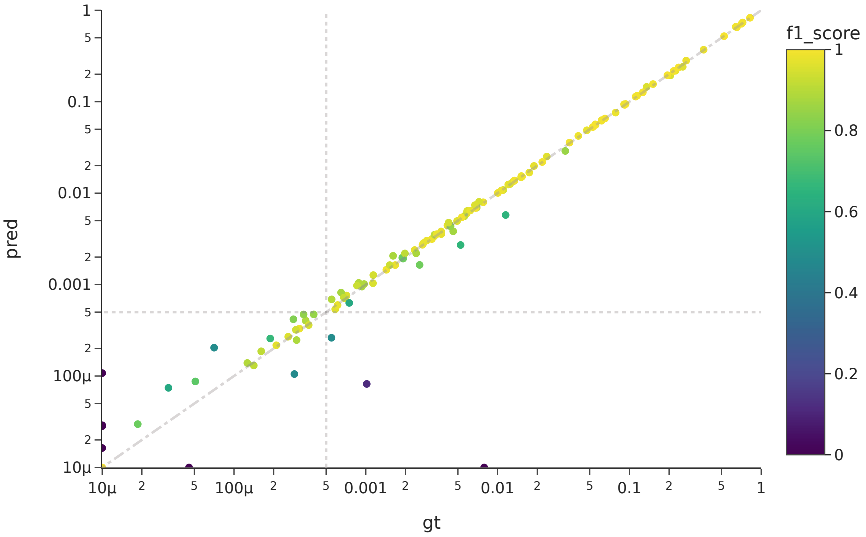


Figure A.1: F_1 scores and predicted MRD values for vie experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

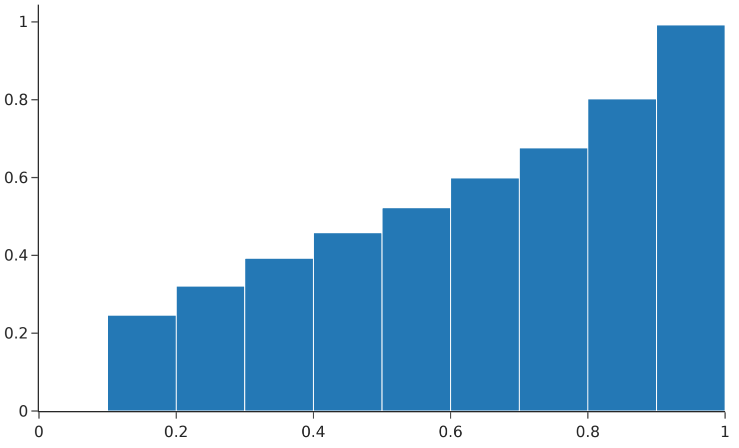


Figure A.2: Calibration of the vie experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

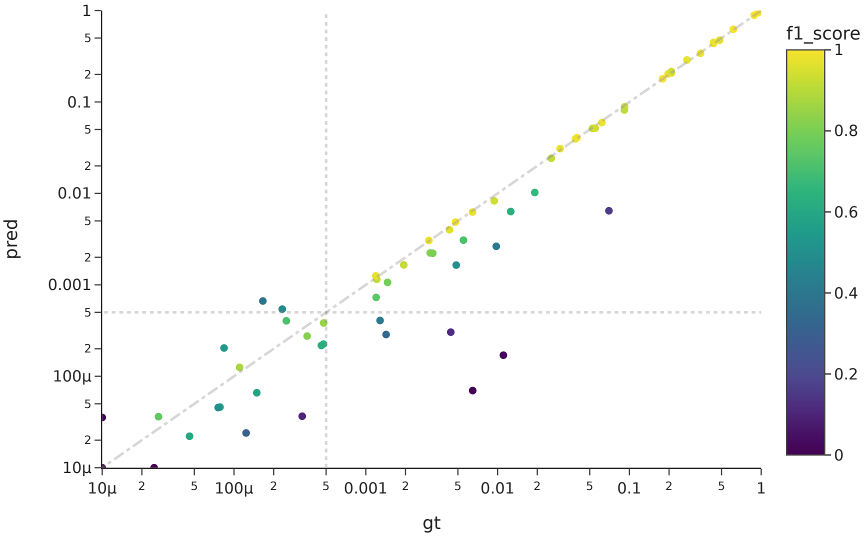


Figure A.3: F_1 scores and predicted MRD values for bln_bue experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

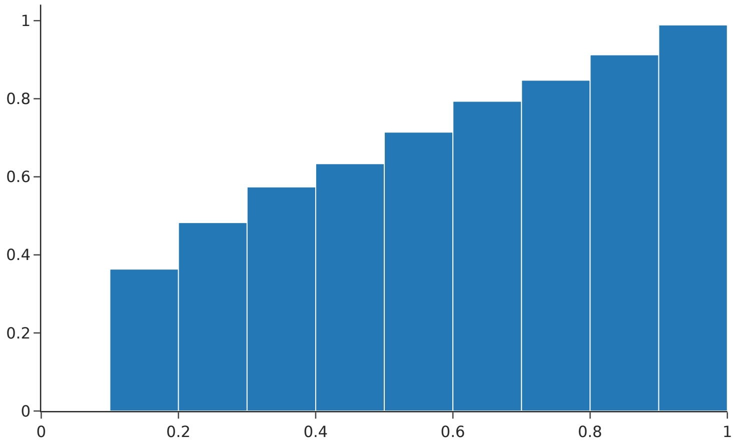


Figure A.4: Calibration of the bln_bue experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

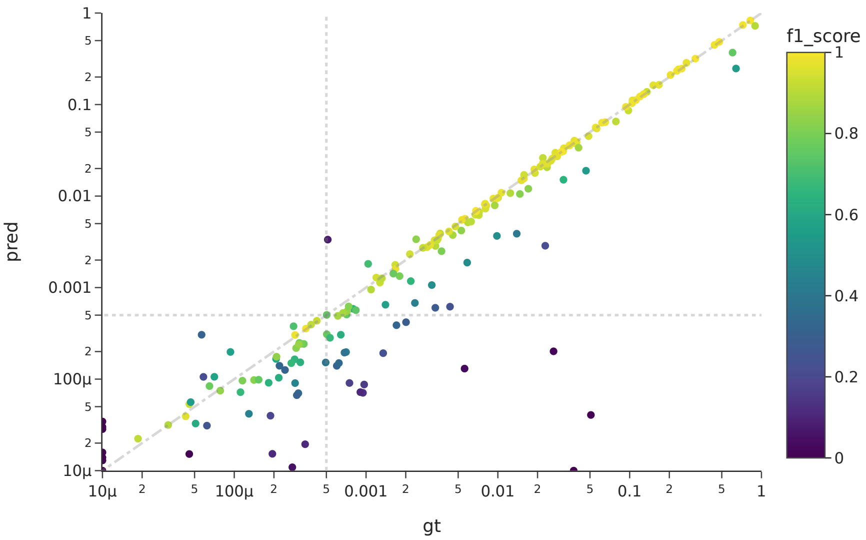


Figure A.5: F_1 scores and predicted MRD values for bln_vie14 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

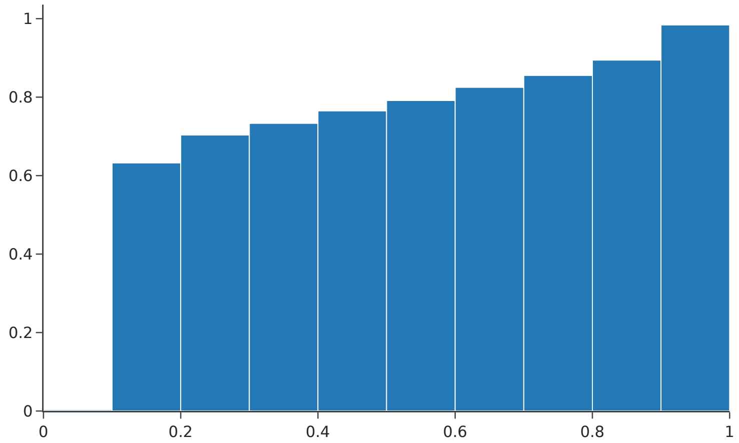


Figure A.6: Calibration of the bln_vie14 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

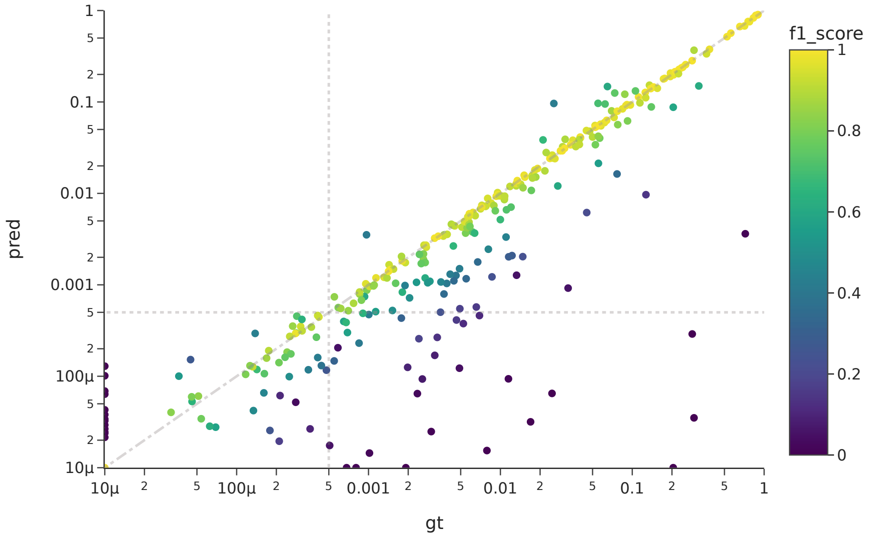


Figure A.7: F_1 scores and predicted MRD values for bln_vie16-20 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

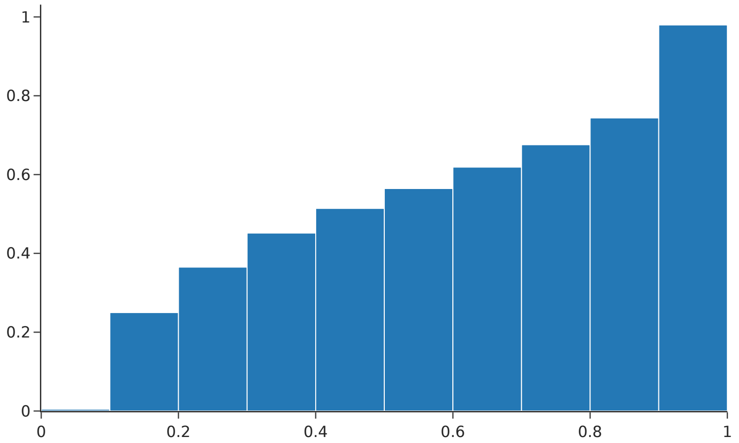


Figure A.8: Calibration of the bln_vie16-20 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

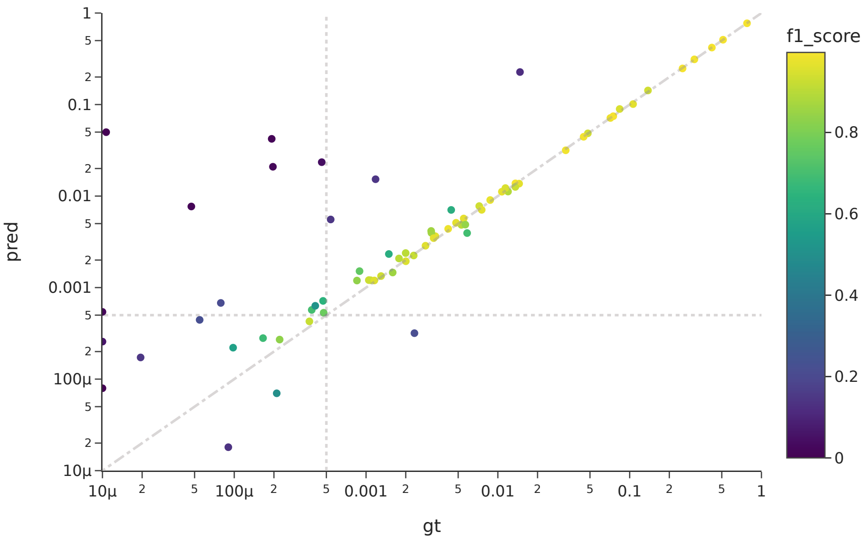


Figure A.9: F_1 scores and predicted MRD values for bue_bln experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

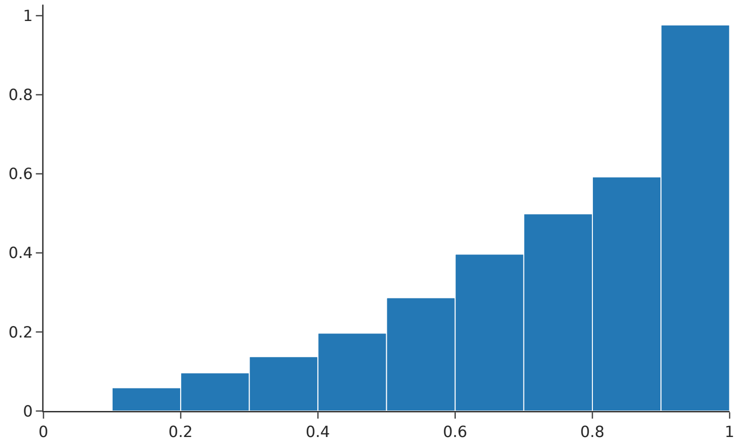


Figure A.10: Calibration of the bue_bln experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

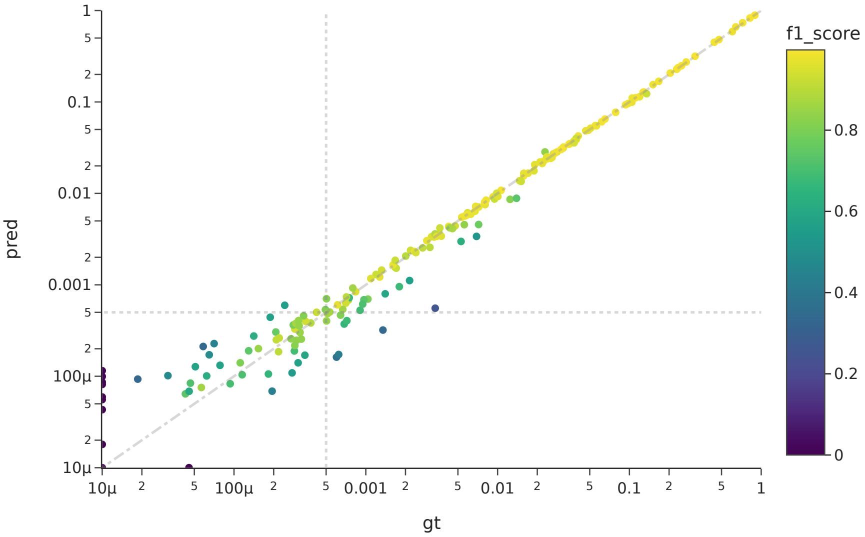


Figure A.11: F_1 scores and predicted MRD values for bue_vie14 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

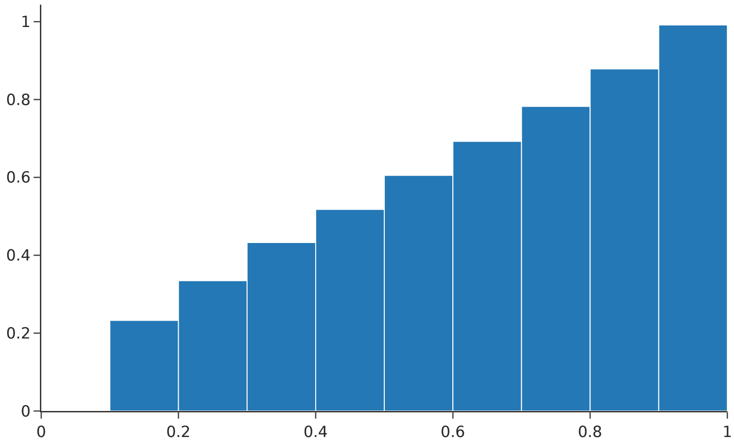


Figure A.12: Calibration of the bue_vie14 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

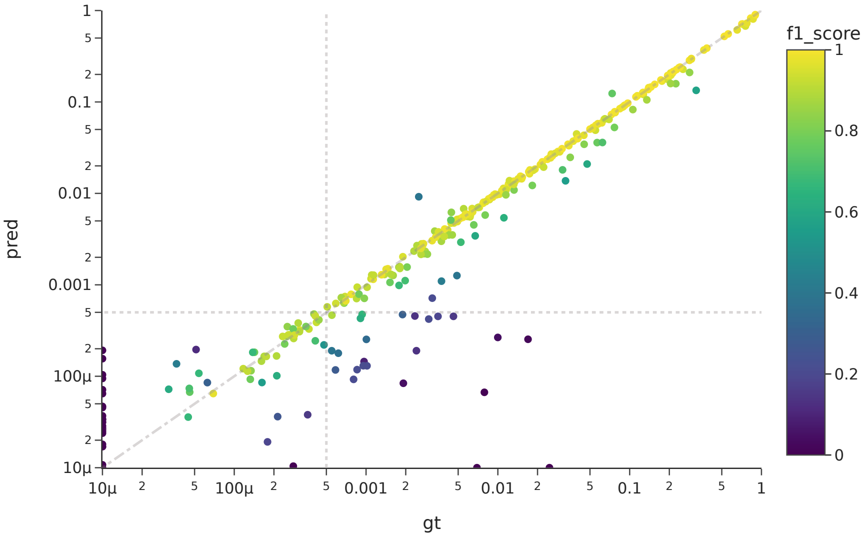


Figure A.13: F_1 scores and predicted MRD values for bue_vie16-20 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

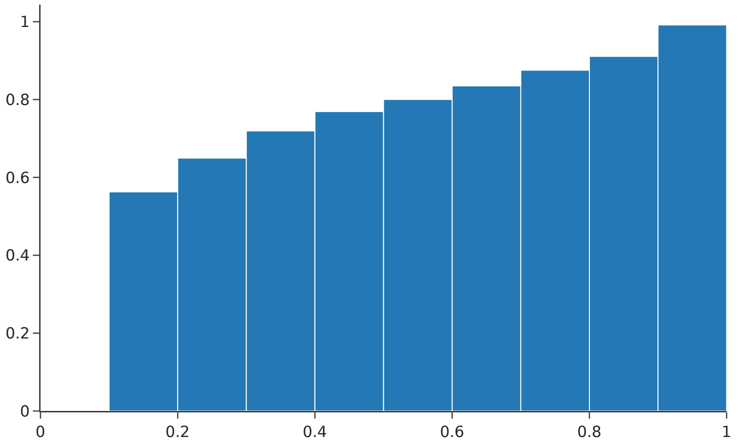


Figure A.14: Calibration of the bue_vie16-20 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

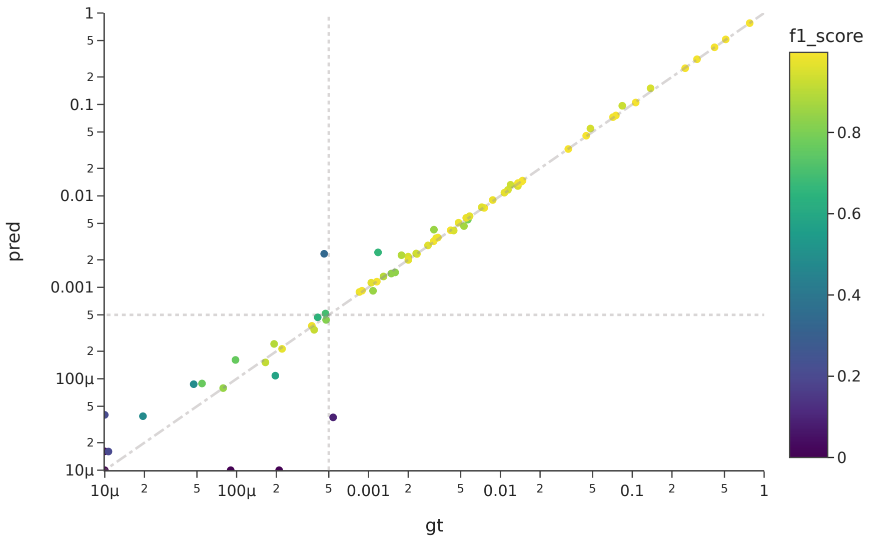


Figure A.15: F_1 scores and predicted MRD values for vie14_bln experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

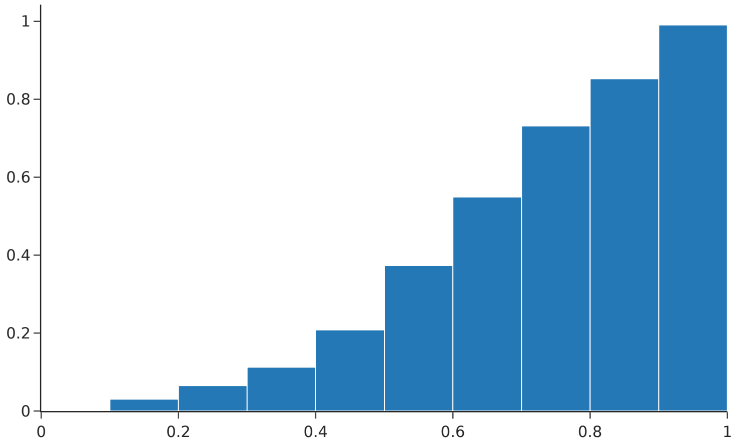


Figure A.16: Calibration of the vie14_bln experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

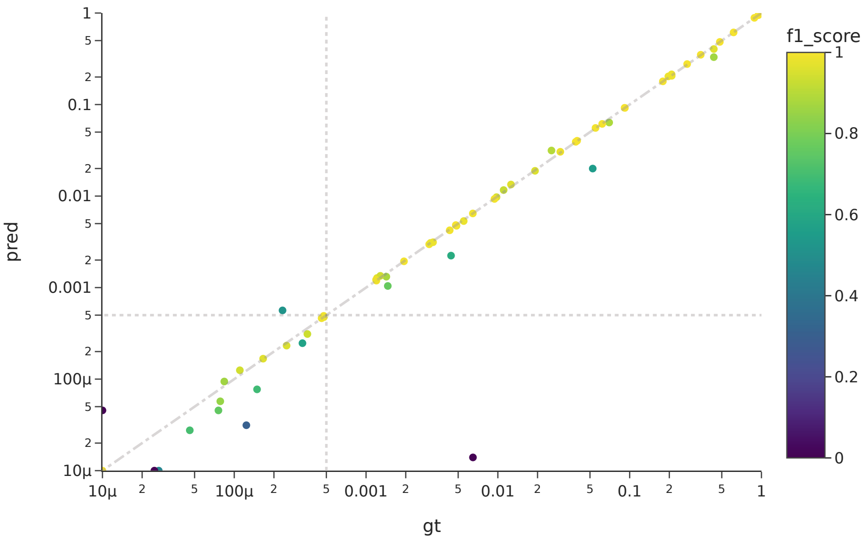


Figure A.17: F_1 scores and predicted MRD values for vie14_bue experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

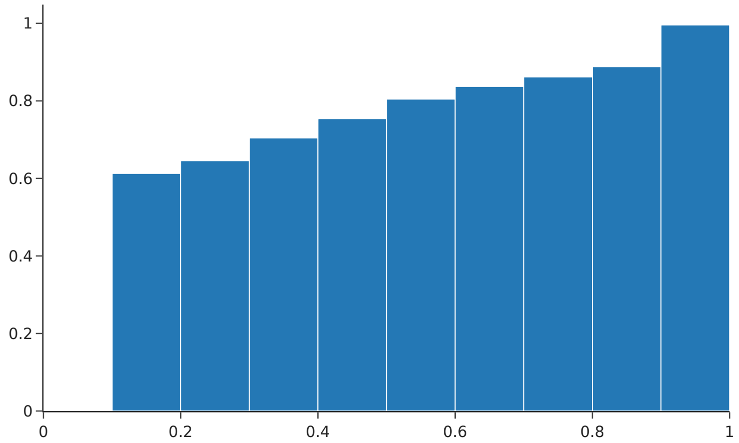


Figure A.18: Calibration of the vie14_bue experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

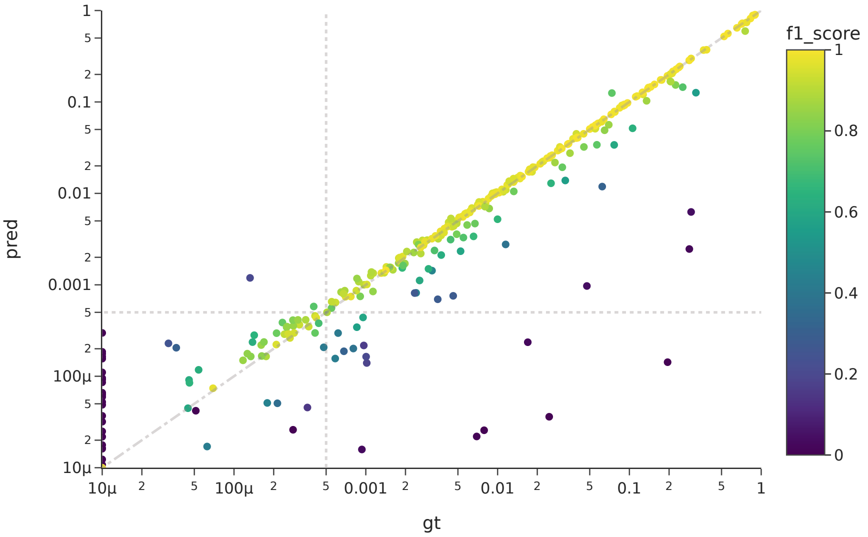


Figure A.19: F_1 scores and predicted MRD values for vie14_vie16-20 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

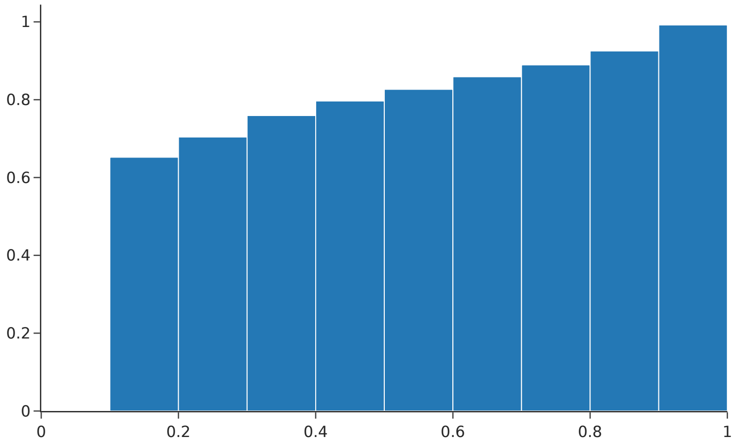


Figure A.20: Calibration of the vie14_vie16-20 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

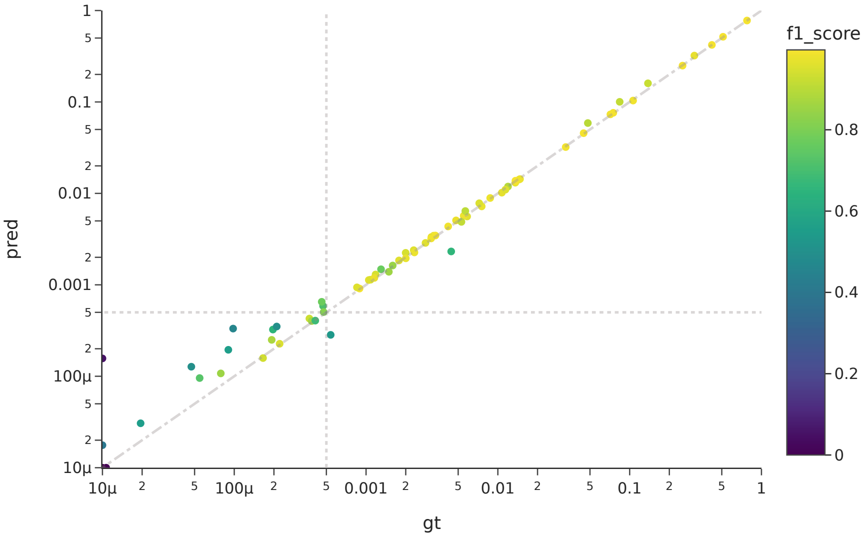


Figure A.21: F_1 scores and predicted MRD values for vie16-20_bln experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

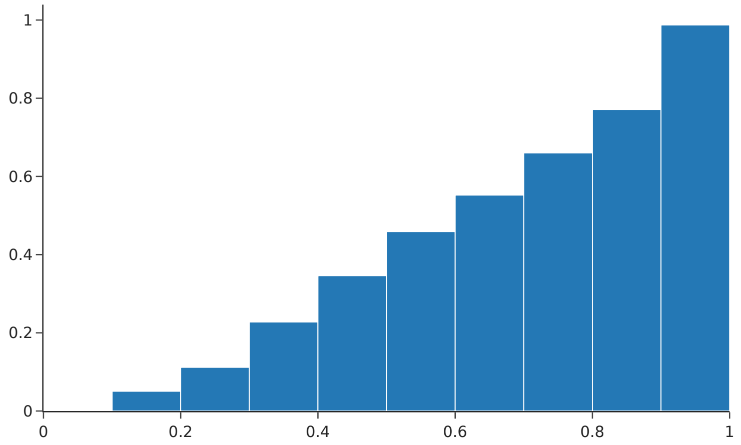


Figure A.22: Calibration of the vie16-20_bln experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

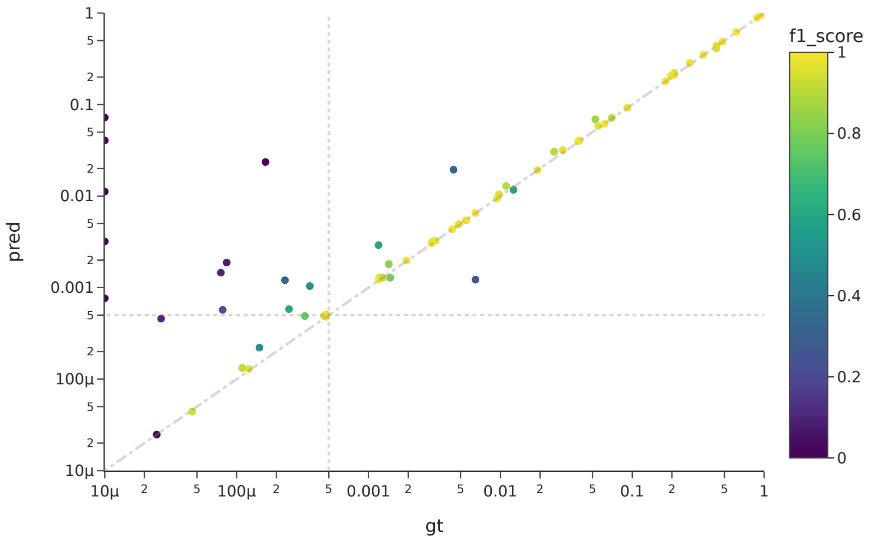


Figure A.23: F_1 scores and predicted MRD values for vie16-20_bue experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

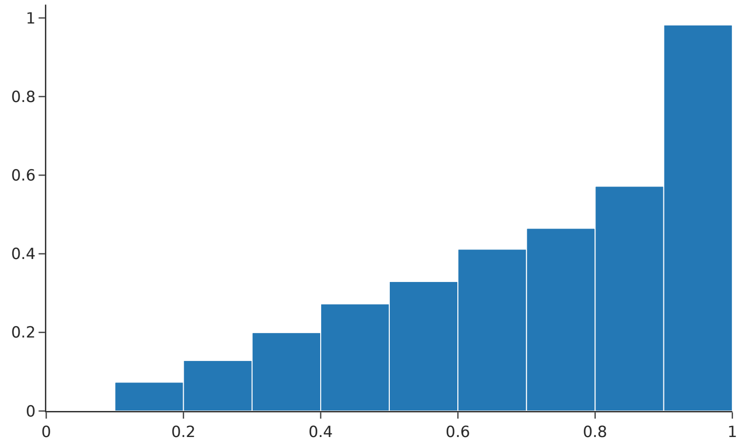


Figure A.24: Calibration of the vie16-20_bue experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

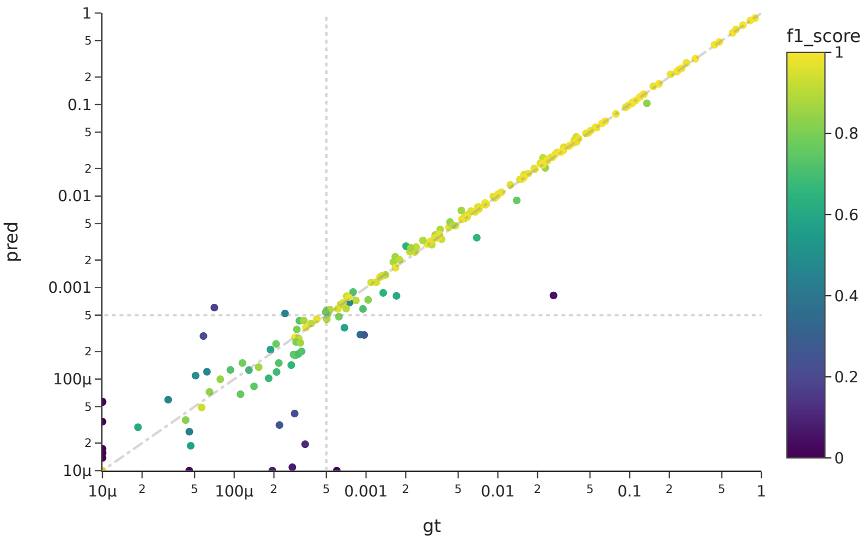


Figure A.25: F_1 scores and predicted MRD values for vie16-20_vie14 experiment. The colors show the F_1 score and the dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification.

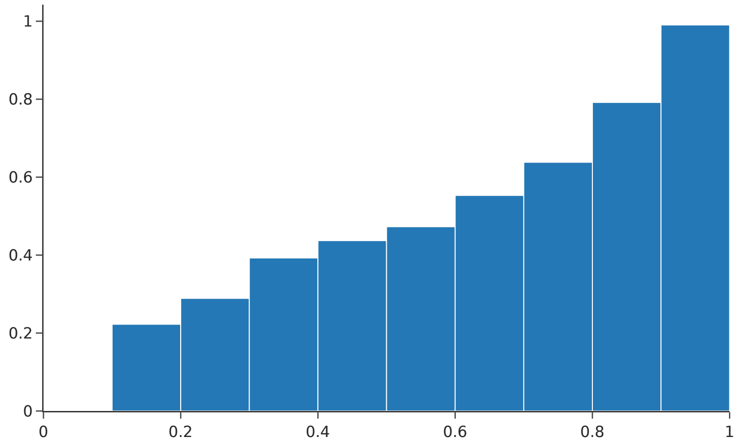


Figure A.26: Calibration of the vie16-20_vie14 experiment. The bar height corresponds to the fraction of events, with a predicted probability in the given range, that correspond to true predictions. For a perfectly calibrated model the histogram would follow a linear trend with 5% for the left most and 95% for the right most bar.

APPENDIX B

Qualitative Stereo Image Compression Results

B.1 SASIC

This section provides additional qualitative results for examples from the InStereo2k dataset in Fig. B.1 and for the Cityscapes dataset in Fig. B.2. The example images always show the right image of the stereo image pair.

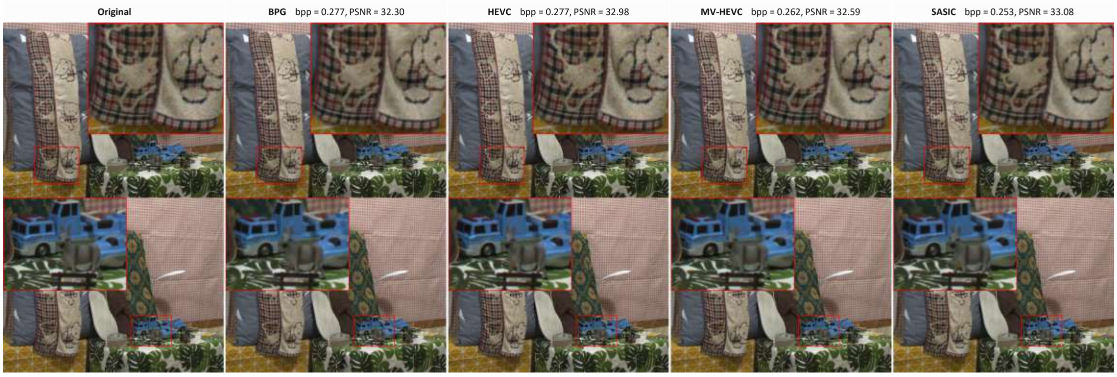

Original
BPG bpp = 0.277, PSNR = 32.30
HEVC bpp = 0.277, PSNR = 32.98
MV-HEVC bpp = 0.262, PSNR = 32.59
SASIC bpp = 0.253, PSNR = 33.08

Figure B.1: A qualitative comparison on an image from the InStereo2K test set.

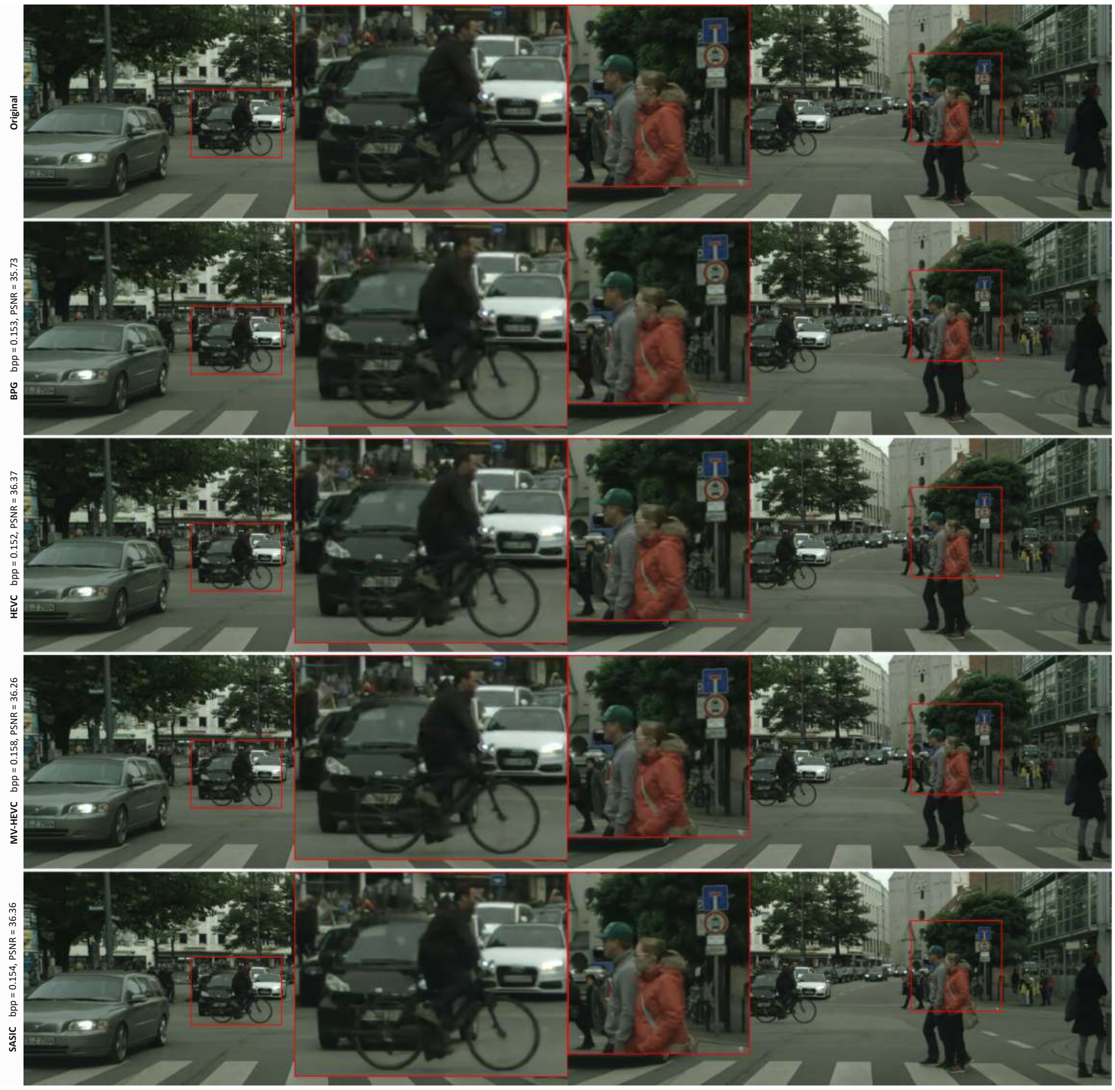

Original
BPG bpp = 0.153, PSNR = 35.73
HEVC bpp = 0.152, PSNR = 36.37
MV-HEVC bpp = 0.158, PSNR = 36.26
SASIC bpp = 0.154, PSNR = 36.36

Figure B.2: A qualitative comparison on an image from the Cityscapes test set. We show the same image in both columns with two different zoomed-out regions.

B.2 ECSIC

This section provides additional qualitative results for examples from the InStereo2k dataset in Fig. B.3 and for Cityscapes in Fig. B.4 and Fig. B.5. The example images always show the right image of the stereo image pair.

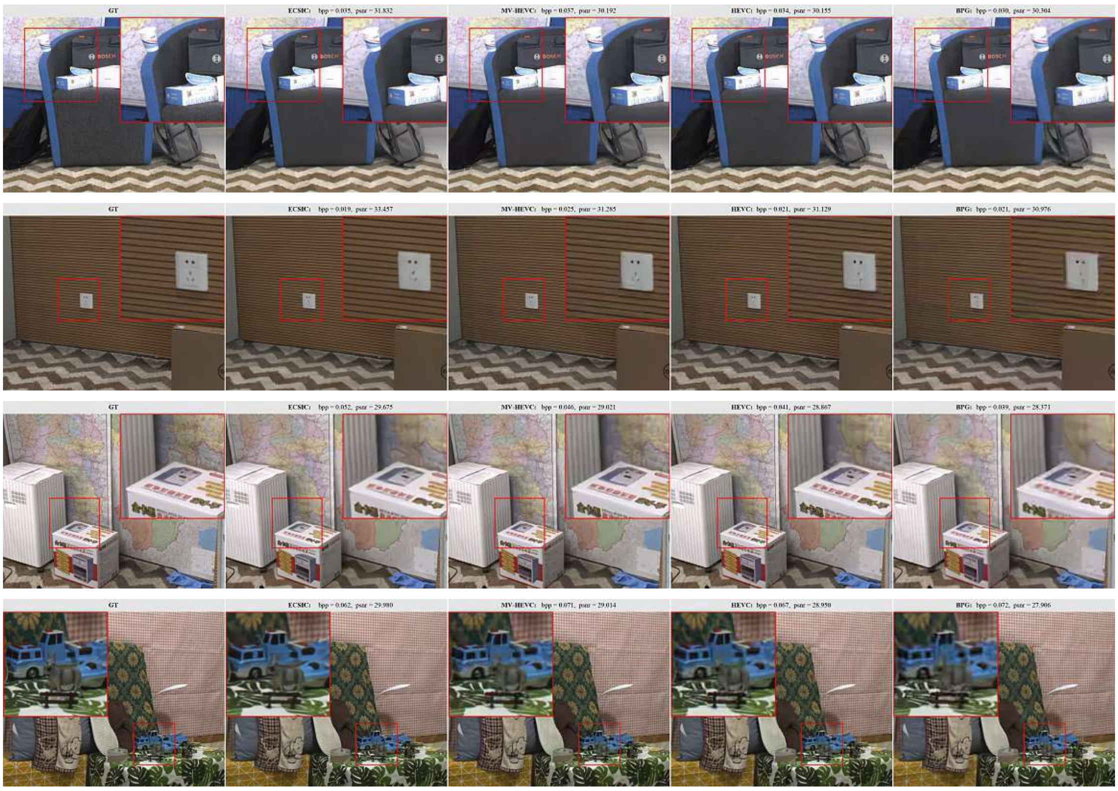

GT
ECSIC: bpp = 0.035, psnr = 31.832
MV-HEVC: bpp = 0.037, psnr = 30.192
HEVC: bpp = 0.034, psnr = 30.155
BPG: bpp = 0.030, psnr = 30.304
GT
ECSIC: bpp = 0.019, psnr = 33.457
MV-HEVC: bpp = 0.025, psnr = 31.285
HEVC: bpp = 0.021, psnr = 31.129
BPG: bpp = 0.021, psnr = 30.976
GT
ECSIC: bpp = 0.052, psnr = 29.675
MV-HEVC: bpp = 0.046, psnr = 29.021
HEVC: bpp = 0.041, psnr = 28.867
BPG: bpp = 0.039, psnr = 28.371
GT
ECSIC: bpp = 0.062, psnr = 29.980
MV-HEVC: bpp = 0.071, psnr = 29.014
HEVC: bpp = 0.067, psnr = 28.950
BPG: bpp = 0.072, psnr = 27.906

Figure B.3: A qualitative comparison on images from the InStereo2K test set.

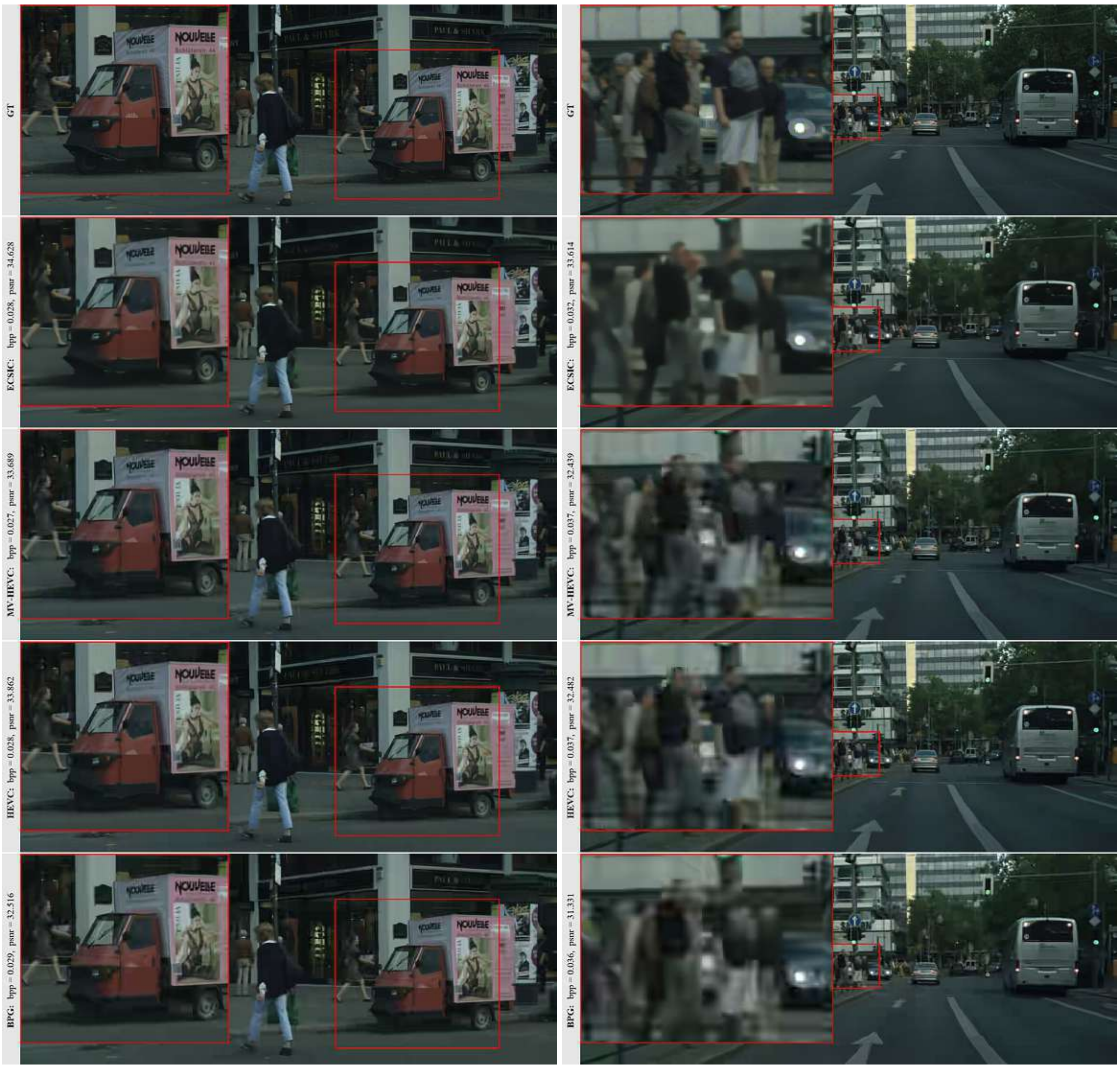

GT
ECSIC: bpp = 0.028, psnr = 34.628
MV-HEVC: bpp = 0.027, psnr = 33.689
HEVC: bpp = 0.028, psnr = 33.862
BPG: bpp = 0.029, psnr = 32.516
GT
ECSIC: bpp = 0.032, psnr = 33.614
MV-HEVC: bpp = 0.037, psnr = 32.439
HEVC: bpp = 0.037, psnr = 32.482
BPG: bpp = 0.036, psnr = 31.331
NOUVELLE

Figure B.4: A qualitative comparison on images from the Cityscapes test set.

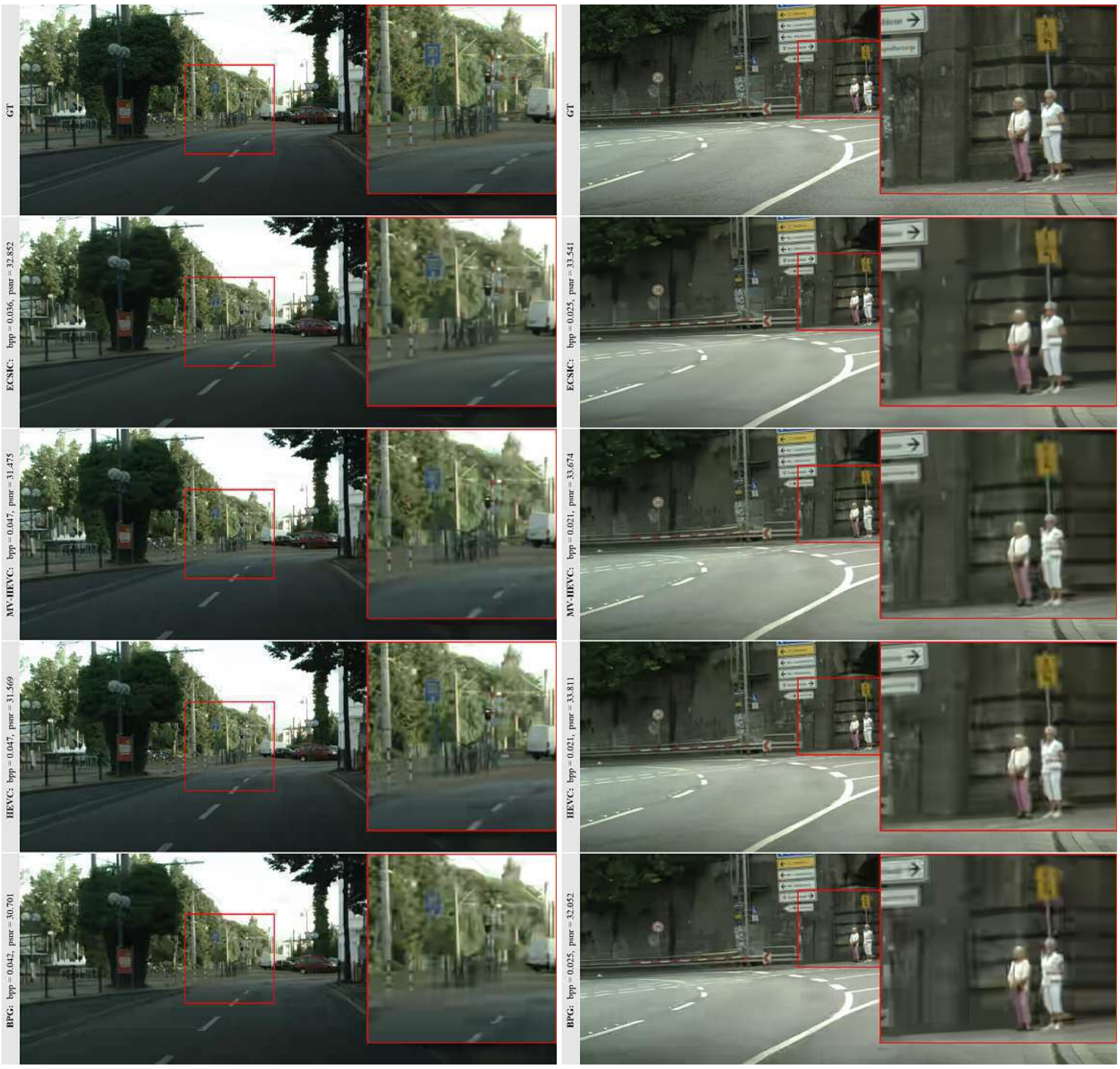

GT
ECSIC: bpp = 0.036, psnr = 32.852
MV-HEVC: bpp = 0.047, psnr = 31.475
HEVC: bpp = 0.047, psnr = 31.569
BPG: bpp = 0.042, psnr = 30.701
GT
ECSIC: bpp = 0.025, psnr = 33.541
MV-HEVC: bpp = 0.021, psnr = 33.674
HEVC: bpp = 0.021, psnr = 33.811
BPG: bpp = 0.025, psnr = 32.052

Figure B.5: A qualitative comparison on images from the Cityscapes test set.

Bibliography

[AAA+23] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[AFH+13] Nima Aghaeepour, Greg Finak, Holger Hoos, Tim R Mosmann, Ryan Brinkman, Raphael Gottardo, and Richard H Scheuermann. Critical assessment of automated flow cytometry data analysis techniques. *Nature methods*, 10(3):228–238, 2013.

[All22] Alliance for Open Media. AVIF image format. `https://aomediacodec.github.io/av1-avif`, 2022. Accessed: 2023-03.

[ATM+19] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 221–231, 2019.

[BCB15] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2015.

[BDKES18] Berat Barakat, Ahmad Droby, Majeed Kassis, and Jihad El-Sana. Text line segmentation for challenging handwritten document images using fully convolutional network. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 374–379. IEEE, 2018.

[Bel14] Fabrice Bellard. BPG Image format. `https://bellard.org/bpg`, 2014. Accessed: 2021-09-24.

[Bjo01a] Gisle Bjontegaard. Calculation of average PSNR differences between RD-curves. *VCEG-M33*, 2001.

[Bjo01b] Gisle Bjontegaard. Calculation of average psnr differences between rd-curves. *VCEG-M33*, 2001.

[BKH16] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[BKWS23] Alexander Bayerl, Manuel Keglevic, Matthias Wödlinger, and Robert Sablatnig. Impact of learned domain specific compression on satellite image object classification. In *26th Computer Vision Winter Workshop (CVWW)*, 2023.

[BLS17] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. In *5th International Conference on Learning Representations, ICLR 2017*, 2017.

[BMK15] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. In *ICLR (Poster)*, 2015.

[BMR$^+$20a] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[BMR$^+$20b] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[BMS$^+$18] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations*, 2018.

[BWX$^+$20] Wei Bao, Wei Wang, Yuhua Xu, Yulan Guo, Siyu Hong, and Xiaohu Zhang. Instereo2k: a large real dataset for stereo matching in indoor scenes. *Science China Information Sciences*, 63(11):1–11, 2020.

[BWY$^+$21] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J. Sullivan, and Jens-Rainer Ohm. Overview of the Versatile Video Coding (VVC) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2021.

[Cam10] Dario Campana. Minimal residual disease in acute lymphoblastic leukemia. *Hematology*, 2010(1):7–12, 2010.

[CLD$^+$20] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. In *International Conference on Learning Representations*, 2020.

[CLLM20] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

[CMS+20] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[CMV+19] Manuel Carbonell, Joan Mas, Mauricio Villegas, Alicia Fornés, and Josep Lladós. End-to-end handwritten text detection and transcription in full pages. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 5, pages 29–34. IEEE, 2019.

[COR+16] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[CSTK20] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7936–7945, 2020.

[CZX+17] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5659–5667, 2017.

[Dao23] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

[DAYR14] Murat Dundar, Ferit Akova, Halid Z Yerebakan, and Bartek Rajwa. A nonparametric bayesian model for joint cell clustering and cluster matching: identification of anomalous sample phenotypes with random effects. *BMC bioinformatics*, 15(1):1–15, 2014.

[DBK+20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[DD95] Robert Desimone and John Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995.

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[DFE+22] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

[DFP+02] Michael N Dworzak, Gertraud Fröschl, Dieter Printz, Georg Mann, Ulrike Pötschger, Nora Mühlegger, Gerhard Fritsch, and Helmut Gadner. Prognostic significance and modalities of flow cytometric minimal residual disease detection in childhood acute lymphoblastic leukemia. *Blood, The Journal of the American Society of Hematology*, 99(6):1952–1958, 2002.

[DGR+08] Michael Norbert Dworzak, Giuseppe Gaipa, Richard Ratei, Marinella Veltroni, Angela Schumich, Oscar Maglia, Leonid Karawajew, Allessandra Benetello, Ulrike Pötschger, Zvenyslava Husak, et al. Standardization of flow cytometric minimal residual disease evaluation in acute lymphoblastic leukemia: Multicentric assessment is feasible. *Cytometry Part B: Clinical Cytometry: The Journal of the International Society for Analytical Cytology*, 74(6):331–340, 2008.

[DKF+17] Markus Diem, Florian Kleber, Stefan Fiel, Tobias Grüning, and Basilis Gatos. cbad: Icdar2017 competition on baseline detection. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 1355–1360. IEEE, 2017.

[DKSG19] Markus Diem, Florian Kleber, Robert Sablatnig, and Basilis Gatos. cbad: Icdar2019 competition on baseline detection. In *2019 15th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 1494–1499. IEEE, 2019.

[DSS22] Philipp Dufter, Martin Schmitt, and Hinrich Schütze. Position information in transformers: An overview. *Computational Linguistics*, 48(3):733–763, 2022.

[DYY+21] Xin Deng, Wenzhe Yang, Ren Yang, Mai Xu, Enpeng Liu, Qianhan Feng, and Radu Timofte. Deep homography for efficient stereo image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1492–1501, June 2021.

[EKB+17] Philipp Eulenberg, Niklas Köhler, Thomas Blasi, Andrew Filby, Anne E Carpenter, Paul Rees, Fabian J Theis, and F Alexander Wolf. Reconstructing cell cycle and disease progression using deep learning. *Nature communications*, 8(1):1–6, 2017.

[FLMS18] Michael Fink, Thomas Layer, Georg Mackenbrock, and Michael Sprinzl. Baseline detection in historical documents using convolutional u-nets. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, pages 37–42. IEEE, 2018.

[GDG+15] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pages 1462–1471. PMLR, 2015.

[GLS+19] Tobias Grüning, Gundram Leifert, Tobias Strauß, Johannes Michael, and Roger Labahn. A two-stage method for text line detection in historical documents. *International Journal on Document Analysis and Recognition (IJDAR)*, 22(3):285–302, 2019.

[GMH13] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013.

[GPAM+20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[GPW+23] Noor Fathima Ghouse, Jens Petersen, Auke Wiggers, Tianlin Xu, and Guillaume Sautière. Neural image compression with a diffusion-based decoder, 2023.

[Gra13] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

[GSBL20] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*, 2020.

[GSG+21] Shangyin Gao, Yibo Shi, Tiansheng Guo, Zhongying Qiu, Yunying Ge, Ze Cui, Yihui Feng, Jing Wang, and Bo Bai. Perceptual learned image compression with continuous rate adaptation. *4th Challenge on Learned Image Compression*, 2(3), 2021.

[GYP+21] Ge Gao, Pei You, Rong Pan, Shunyuan Han, Yuanyuan Zhang, Yuchao Dai, and Hojae Lee. Neural image compression via attentional multi-scale back projection and frequency decomposition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14677–14686, October 2021.

[GZFC21a] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. Causal contextual prediction for learned image compression. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2021.

[GZFC21b] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen. Soft then hard: Rethinking the quantization in neural image compression. In *International Conference on Machine Learning*, pages 3920–3929. PMLR, 2021.

[HBM+22] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[HCH+19] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520. IEEE, 2019.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[HSD+21] Zihao Huang, Zhe Sun, Feng Duan, Andrzej Cichocki, Peiying Ruan, and Chao Li. L3c-stereo: Lossless compression for stereo images. *arXiv preprint arXiv:2108.09422*, 2021.

[HWC+22] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022.

[HWS22] Thomas Heitzinger, Matthias Woedlinger, and David G Stork. Artist-specific style transfer for semantic segmentation of paintings: The value of large corpora of surrogate artworks. *Electronic Imaging*, 34(13):186–1, 2022.

[HXW+21] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, 34:15908–15919, 2021.

[HYP+22] Dailan He, Ziming Yang, Weikun Peng, Rui Ma, Hongwei Qin, and Yan Wang. ELIC: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), pages 5718–5727, June 2022.

[HYY+22] Dailan He, Ziming Yang, Hongjiu Yu, Tongda Xu, Jixiang Luo, Yuan Chen, Chenjian Gao, Xinjie Shi, Hongwei Qin, and Yan Wang. PO-ELIC: Perception-oriented efficient learned image coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1764–1769, June 2022.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[HZS+21] Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14771–14780, June 2021.

[IAB+21] Muhammad Shahid Iqbal, Iftikhar Ahmad, Luo Bin, Suleman Khan, and Joel JPC Rodrigues. Deep learning recognition of diseased and normal cell representation. *Transactions on Emerging Telecommunications Technologies*, 32(7):e4017, 2021.

[IKN98] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998.

[JGB+21] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021.

[JSF+20] Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The break-even point on optimization trajectories of deep neural networks. *arXiv preprint arXiv:2002.09572*, 2020.

[JSZ+15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[JWF16] Kerstin Johnsson, Jonas Wallin, and Magnus Fontes. Bayesflow: latent modeling of flow cytometry cell populations. *BMC bioinformatics*, 17(1):1–16, 2016.

[KB14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[KGB+22] A Burakhan Koyuncu, Han Gao, Atanas Boev, Georgii Gaikov, Elena Alshina, and Eckehard Steinbach. Contextformer: A transformer with spatio-channel attention for context modeling in learned image compression. In *European Conference on Computer Vision*, pages 447–463. Springer, 2022.

[KHL22] Jun-Hyuk Kim, Byeongho Heo, and Jong-Seok Lee. Joint global and local hierarchical priors for learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5992–6001, June 2022.

[KKL19] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2019.

[KMGW+24] Florian Kowarsch, Margarita Maurer-Granofszky, Lisa Weijler, Matthias Wödlinger, Michael Reiter, Angela Schumich, Tamar Feuerstein, Simona Sala, Michaela Nováková, Giovanni Faggin, et al. Fcm marker importance for mrd assessment in t-cell acute lymphoblastic leukemia: An aieop-bfm-all-flow study group report. *Cytometry Part A*, 105(1):24–35, 2024.

[KMH+20] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[KVPF20] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020.

[KWK23] Jan Kotera, Matthias Wödlinger, and Manuel Keglevic. Learned lossy image compression for volumetric medical data. In *26th Computer Vision Winter Workshop (CVWW)*, 2023.

[KWW+22] Florian Kowarsch, Lisa Weijler, Matthias Wödlinger, Michael Reiter, Margarita Maurer-Granofszky, Angela Schumich, Elisa O Sajaroff, Stefanie Groeneveld-Krentz, Jorge G Rossi, Leonid Karawajew, et al. Towards self-explainable transformers for cell classification in flow cytometry data. In *International Workshop on Interpretability of Machine Intelligence in Medical Image Computing*, pages 22–32. Springer, 2022.

[KWW+23] Florian Kowarsch, Lisa Magdalena Weijler, Matthias Gerold Wödlinger, Florian Kleber, Margarita Maurer-Granofszky, Michael Reiter, and Michael Dworzak. Explainable visualization techniques for transformers in flow cytometry data. [Conference Presentation]. 26th Computer Vision Winter Workshop (CVWW), 2023.

[LCD+18] Yuqian Li, Bruno Cornelis, Alexandra Dusa, Geert Vanmeerbeeck, Dries Vercruysse, Erik Sohn, Kamil Blaszkiewicz, Dimiter Prodanov, Peter Schelkens, and Liesbet Lagae. Accurate label-free 3-part leukocyte recognition with single cell lens-free imaging flow cytometry. *Computers in biology and medicine*, 96:147–156, 2018.

[LCG+19] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

[LH16] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[LHL+22] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[LLC+21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[LLK+19a] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pages 3744–3753. PMLR, 2019.

[LLK+19b] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, 2019.

[LMB+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco:

Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[LMW+22] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022.

[LOG+19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[LSR+18] Roxane Licandro, Thomas Schlegl, Michael Reiter, Markus Diem, Michael Dworzak, Angela Schumich, Georg Langs, and Martin Kampel. Wgan latent space embeddings for blast identification in childhood acute myeloid leukaemia. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3868–3873. IEEE, 2018.

[LSS+17] Huamin Li, Uri Shaham, Kelly P Stanton, Yi Yao, Ruth R Montgomery, and Yuval Kluger. Gating mass cytometry data by deep learning. *Bioinformatics*, 33(21):3423–3430, 2017.

[LWU19a] Jerry Liu, Shenlong Wang, and R. Urtasun. DSIC: Deep stereo image compression. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3136–3145, 2019.

[LWU19b] Jerry Liu, Shenlong Wang, and Raquel Urtasun. Dsic: Deep stereo image compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3136–3145, 2019.

[LXT+18] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[LYZ+23] Xiaoran Liu, Hang Yan, Shuo Zhang, Chenxin An, Xipeng Qiu, and Dahua Lin. Scaling laws of rope-based extrapolation. *arXiv preprint arXiv:2310.05209*, 2023.

[MAT+18] Fabian Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Gool. Conditional probability models for deep image compression. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4394–4402, 2018.

[MBT18] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. *Advances in Neural Information Processing Systems*, 31:10771–10780, 2018.

[MHGK14] Volodymyr Mnih, Nicolas Manfred Otto Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *NIPS*, 2014.

[MKW17] Bastien Moysset, Christopher Kermorvant, and Christian Wolf. Full-page text recognition: Learning where to start and when to stop. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 871–876. IEEE, 2017.

[MMSW06] Philipp Merkle, Karsten Muller, Aljoscha Smolic, and Thomas Wiegand. Efficient compression of multi-view video exploiting inter-view dependencies based on h. 264/mpeg4-avc. In *2006 IEEE International Conference on Multimedia and Expo*, pages 1717–1720. IEEE, 2006.

[MÖGG23] Nitish Mital, Ezgi Özyilkan, Ali Garjani, and Deniz Gündüz. Neural distributed image compression with cross-attention feature alignment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2498–2507, 2023.

[MS20] David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3339–3343, 2020.

[MTTA20] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 11913–11924. Curran Associates, Inc., 2020.

[NDBS20] Noga Nissim, Matan Dudaie, Itay Barnea, and Natan T Shaked. Real-time stain-free classification of cancer cells and blood cells using interferometric phase microscopy and machine learning. *Cytometry Part A*, 2020.

[NDR+14] Iftekhar Naim, Suprakash Datta, Jonathan Rebhahn, James S Cavenaugh, Tim R Mosmann, and Gaurav Sharma. Swift—scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, part 1: Algorithm design. *Cytometry Part A*, 85(5):408–421, 2014.

[NZY21] Zhaoyang Niu, Guoqiang Zhong, and Hui Yu. A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62, 2021.

[ODO16] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10):e3, 2016.

[Ope23] R OpenAI. Gpt-4 technical report. *arXiv*, pages 2303–08774, 2023.

[PFBK21] Shi Pan, Chris Finlay, Chri Besenbruch, and William Knottenbelt. Three gaps for quantisation in learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 720–726, 2021.

[PK22] Namuk Park and Songkuk Kim. How do vision transformers work? In *International Conference on Learning Representations*, 2022.

[PRL08] Ching-Hon Pui, Leslie L Robison, and A Thomas Look. Acute lymphoblastic leukaemia. *The Lancet*, 371(9617):1030–1043, 2008.

[PZY$^+$17] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4353–4361, 2017.

[QSL$^+$22] Yichen Qian, Xiuyu Sun, Ming Lin, Zhiyu Tan, and Rong Jin. Entroformer: A transformer-based entropy model for learned image compression. In *International Conference on Learning Representations*, 2022.

[QTS$^+$20] Yichen Qian, Zhiyu Tan, Xiuyu Sun, Ming Lin, Dongyang Li, Zhenhong Sun, Hao Li, and Rong Jin. Learning accurate entropy model with global reference for image compression, 2020.

[RDS$^+$19] Michael Reiter, Markus Diem, Angela Schumich, Margarita Maurer-Granofszky, Leonid Karawajew, Jorge G Rossi, Richard Ratei, Stefanie Groeneveld-Krentz, Elisa O Sajaroff, Susanne Suhendra, et al. Automated flow cytometric mrd assessment in childhood acute b-lymphoblastic leukemia using supervised machine learning. *Cytometry Part A*, 95(9):966–975, 2019.

[RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention (MICCAI)*, pages 234–241. Springer, 2015.

[RKH$^+$21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[RPV+19] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. *Advances in neural information processing systems*, 32, 2019.

[RRK+16] Michael Reiter, Paolo Rota, Florian Kleber, Markus Diem, Stefanie Groeneveld-Krentz, and Michael Dworzak. Clustering of cell populations in flow cytometry data using a combination of gaussian mixtures. *Pattern Recognition*, 60:1029–1040, 2016.

[RSL+20] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, et al. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2020.

[RUK+21] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34:12116–12128, 2021.

[RWC+19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[SAL+24] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[SBD+15] Till Sörensen, Sabine Baumgart, Pawel Durek, Andreas Grützkau, and Thomas Häupl. immunoclust—an automated analysis pipeline for the identification of immunophenotypic signatures in high-dimensional cytometric datasets. *Cytometry Part A*, 87(7):603–615, 2015.

[SCE01] Athanassios Skodras, Charilaos Christopoulos, and Touradj Ebrahimi. The jpeg 2000 still image compression standard. *IEEE Signal processing magazine*, 18(5):36–58, 2001.

[SDCW19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[SHM+18] Patrick Schone, Christian Hargraves, Jon Morrey, Rachael Day, and Mindy Jacox. Neural text line segmentation of multilingual print and handwriting with recognition-based evaluation. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 265–272. IEEE, 2018.

[SLR$^+$19] Jakob Scheithe, Roxane Licandro, Paolo Rota, Michael Reiter, Markus Diem, and Martin Kampel. Monitoring acute lymphoblastic leukemia therapy with stacked denoising autoencoders. In *Computer Aided Intervention and Diagnostics in Clinical and Medical Images*, pages 189–197. Springer, 2019.

[SOHW12] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the High Efficiency Video Coding (HEVC) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012.

[SRT$^+$17] Joan Andreu Sánchez, Verónica Romero, Alejandro H Toselli, Mauricio Villegas, and Enrique Vidal. Icdar2017 competition on handwritten text recognition on the read dataset. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 1383–1388. IEEE, 2017.

[SUV18] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of NAACL-HLT*, pages 464–468, 2018.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[TAL18] Michael Tschannen, Eirikur Agustsson, and Mario Lucic. Deep generative models for distribution-preserving lossy compression. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[TBY$^+$19] Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. Transformer dissection: An unified understanding for transformer's attention via the lens of kernel. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2019.

[TCW$^+$95] John K Tsotsos, Scan M Culhane, Winky Yan Kei Wai, Yuzhong Lai, Neal Davis, and Fernando Nuflo. Modeling visual attention via selective tuning. *Artificial intelligence*, 78(1-2):507–545, 1995.

[TDA$^+$20] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena: A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2020.

[TDBM20] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020.

[TOH+16] George Toderici, Sean M O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. *CoRR*, abs/1511.06085, 2016.

[TSCH17] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395*, 2017.

[TSHM22] Lucas Theis, Tim Salimans, Matthew D. Hoffman, and Fabian Mentzer. Lossy compression with gaussian diffusion, 2022.

[TW19] Chris Tensmeyer and Curtis Wigington. Training full-page handwritten text recognition models without annotated line breaks. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1–8. IEEE, 2019.

[vRSL+24] Ties van Rozendaal, Tushar Singhal, Hoang Le, Guillaume Sautiere, Amir Said, Krishna Buska, Anjuman Raha, Dimitris Kalatzis, Hitarth Mehta, Frank Mayer, et al. Mobilenvc: Real-time 1080p neural video compression on a mobile device. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4323–4333, 2024.

[VSP+17a] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[VSP+17b] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[Wal92] G.K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992.

[WGGH18] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018.

[Wil92] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

[WKK+24] Matthias Wödlinger, Jan Kotera, Manuel Keglevic, Jan Xu, and Robert Sablatnig. ECSIC: Epipolar cross attention for stereo image compression. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3436–3445, 2024.

[WKR+24] Lisa Weijler, Florian Kowarsch, Michael Reiter, Pedro Hermosilla, Margarita Maurer-Granofszky, and Michael Dworzak. Fate: Feature-agnostic transformer-based encoder for learning generalized embedding spaces in flow cytometry data. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7956–7964, 2024.

[WKW+22] Lisa Weijler, Florian Kowarsch, Matthias Wödlinger, Michael Reiter, Margarita Maurer-Granofszky, Angela Schumich, and Michael N Dworzak. Umap based anomaly detection for minimal residual disease quantification within acute myeloid leukemia. *Cancers*, 14(4):898, 2022.

[WKXS22] Matthias Wödlinger, Jan Kotera, Jan Xu, and Robert Sablatnig. SASIC: Stereo image compression with latent shifts and stereo attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 661–670, 2022.

[WRW+22a] Matthias Wodlinger, Michael Reiter, Lisa Weijler, Margarita Maurer-Granofszky, Angela Schumich, and Michael N Dworzak. Automated identification of cell populations in flow cytometry data with transformers. *Computers in Biology and Medicine*, page 105314, 2022.

[WRW+22b] Matthias Woedlinger, Michael Reiter, Lisa Weijler, Margarita Maurer-Granofszky, Angela Schumich, Elisa O Sajaroff, Stefanie Groeneveld-Krentz, Jorge G Rossi, Leonid Karawajew, Richard Ratei, et al. Automated identification of cell populations in flow cytometry data with transformers. *Computers in Biology and Medicine*, 144:105314, 2022.

[WS20] Matthias Wödlinger and Robert Sablatnig. Classification and segmentation of scanned library catalogue cards using convolutional neural networks. In *Proceedings of the Joint Austrian Computer Vision and Robotics Workshop 2020*, pages 90–91. Austrian Association for Pattern Recognition (ÖAGM/AAPR), 2020.

[WS21] Matthias Wödlinger and Robert Sablatnig. Text baseline recognition using a recurrent convolutional neural network. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4673–4679. IEEE, 2021.

[WTD+18] Curtis Wigington, Chris Tensmeyer, Brian Davis, William Barrett, Brian Price, and Scott Cohen. Start, follow, read: End-to-end full-page handwriting recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 367–383, 2018.

[WXW+21] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8741–8750, 2021.

[WZA+21] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5463–5474, 2021.

[XBK+15] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[XCC21] Yueqi Xie, Ka Leong Cheng, and Qifeng Chen. Enhanced invertible encoding for learned image compression. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 162–170, New York, NY, USA, 2021. Association for Computing Machinery.

[XGD+17] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1492–1500, 2017.

[XLC+20] Jan Xu, Alexander Lytchier, Ciro Cursio, Dimitrios Kollias, Christian Besenbruch, and Arsalan Zafar. Efficient context-aware lossy image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 130–131, 2020.

[YBM20] Yibo Yang, Robert Bamler, and Stephan Mandt. Improving inference for neural image compression. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 573–584. Curran Associates, Inc., 2020.

[YHG+16] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29, 2016.

[YWW+20] Xinyi Ying, Yingqian Wang, Longguang Wang, Weidong Sheng, Wei An, and Yulan Guo. A stereo attention module for stereo image super-resolution. *IEEE Signal Processing Letters*, 27:496–500, 2020.

[ZGBFF16] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004, 2016.

[ZGD+20] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan

Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297, 2020.

[ZIE+18] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595. IEEE, 2018.

[ZLZ+21] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021.

[ZMH+20] Max Zhao, Nanditha Mallesh, Alexander Höllein, Richard Schabath, Claudia Haferlach, Torsten Haferlach, Franz Elsner, Hannes Lüling, Peter Krawitz, and Wolfgang Kern. Hematologist-level classification of mature b-cell neoplasm using deep learning on multiparameter flow cytometry data. *Cytometry Part A*, 97(10):1073–1080, 2020.

[ZSL+20] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

[ZSZ22] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang. The devil is in the details: Window-based attention for image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17492–17501, June 2022.

[ZSZ23] Xinjie Zhang, Jiawei Shao, and Jun Zhang. LDMIC: Learning-based distributed multi-view image coding. *arXiv preprint arXiv:2301.09799*, 2023.

[ZYC22] Yinhao Zhu, Yang Yang, and Taco Cohen. Transformer-based transform coding. In *International Conference on Learning Representations*, 2022.