

FEDERAL INSTITUTE OF EDUCATION, SCIENCE, AND TECHNOLOGY OF CEARÁ
PRO-RECTORY OF RESEARCH, POST-GRADUATION, AND INNOVATION
POST-GRADUATION PROGRAM IN COMPUTER SCIENCE

Levi Cordeiro Carvalho

**Exploring-exploiting in Deep Embeddings for Image Quality
Retargeting Assessment**

Fortaleza – CE

August 26th, 2024

Levi Cordeiro Carvalho

Exploring-exploiting in Deep Embeddings for Image Quality Retargeting Assessment

Dissertation submitted to the Post-Graduation
Program in Computer Science at the Federal
Institute of Education, Science, and Technology
of Ceará as a partial requirement for the degree
of Master in Computer Science.

Research Area: Artificial Intelligence

Advisor: Saulo Anderson Freitas de Oliveira

Federal Institute of Education, Science, and Technology of Ceará (IFCE)

Pro-Rectorate of Research, Post-Graduation, and Innovation (PRPI)

Post-Graduation Program in Computer Science (PPGCC)

Fortaleza – CE

August 26th, 2024

Dados Internacionais de Catalogação na Publicação
Instituto Federal do Ceará - IFCE
Sistema de Bibliotecas - SIBI
Ficha catalográfica elaborada pelo SIBI/IFCE, com os dados fornecidos pelo(a) autor(a)

- C331e Carvalho, Levi Cordeiro.
 EXPLORING-EXPLOITING IN DEEP EMBEDDINGS FOR IMAGE QUALITY RETARGETING
 ASSESSMENT / Levi Cordeiro Carvalho. - 2024.
 80 f. : il. color.
- Dissertação (Mestrado) - Instituto Federal do Ceará, Mestrado em Ciência da Computação, Campus
 Fortaleza, 2024.
 Orientação: Prof. Dr. Saulo Anderson Freitas de Oliveira.
1. Autoencoder. 2. Avaliação de Qualidade de Imagem Redimensionada. 3. Aprendizagem Profunda. I.
 Titulo.

CDD 004



Federal Institute of Education, Science, and Technology of Ceará (IFCE)
Pro-Rectorcy of Research, Post-Graduation, and Innovation (PRPI)
Post-Graduation Program in Computer Science (PPGCC)

Levi Cordeiro Carvalho

This dissertation has been deemed suitable for the attainment of the title of Master in Computer Science, having been approved by the Coordination of the Post-Graduation Program in Computer Science at the Federal Institute of Education, Science, and Technology of Ceará and by the examination board:

Advisor: Saulo Anderson Freitas de Oliveira
Federal Institute of Education, Science, and Technology
of Ceará (IFCE)

José Wellington Franco da Silva
Federal University of Ceará (UFC)

João Paulo Pordeus Gomes
Federal University of Ceará (UFC)

Leandro Bezerra Marinho
Federal Institute of Education, Science, and Technology
of Ceará (IFCE)

Fortaleza – CE
August 26th, 2024

I dedicate this work to my parents, my primary guides in shaping me as a human being.

Acknowledgements

To my family, for the support they have always given me throughout my master's program and the completion of this work.

To my advisor, Prof. Dr. Saulo Oliveira, for offering his expertise, patience, dedication, and time to the development of this project.

To all my friends, especially those who have been with me since school on countless nights of playing, watching movies, hanging out, or just chatting: Rapha, Tundê, Wesley, Naldo, and company.

I also thank the IFCE Fortaleza campus and its professors for everything I learned and the experiences I gained during the course.

Finally, I am grateful to everyone who contributed, directly or indirectly, to my academic and personal development.

*“Be the person you would like to meet.
If our life is but a spark in the eyes of gods,
let’s blind them at least once.”
Shrine of Believers (Hollow Knight)*

Resumo

O redimensionamento de imagem envolve ajustar imagens de entrada para dimensões arbitrárias (linhas e colunas) enquanto preserva as regiões de interesse. Avaliar a qualidade da imagem sob diferentes proporções de aspecto é particularmente desafiador, pois requer correspondência de conteúdo e análise de conteúdo semântico. Neste estudo, são introduzidos dois algoritmos objetivos de avaliação de qualidade para redimensionamento de imagem. O primeiro é uma abordagem baseada em autoencoder, chamada AutoEncoder Information MaP Similarity (AEIMPS). Ele utiliza o fator de redimensionamento, o espaço latente e a imagem reconstruída em seu pipeline. O segundo, inspirado no primeiro, é o Encoding MaP Image Retargeting Quality (EMPIRQ). É uma abordagem sem referência que avalia a qualidade no redimensionamento de imagem através de arquiteturas de regressores que utilizam codificações calculadas por autoencoders como entrada. A proposta foi avaliada em um conjunto de dados do estado da arte amplamente reconhecidos, onde espectadores humanos forneceram opiniões pessoais sobre a qualidade perceptual. Além disso, também são propostos cinco modelos base para analisar o viés em relação aos resultados. O EMPIRQ obteve resultados notáveis: LLC de 0,98, SRCC de 0,98, RMSE de 3,06 e OR de 0,01, superando todos os modelos base e algoritmos de estado da arte sob comparação.

Palavras-chave: autoencoder. avaliação de qualidade de imagem redimensionada. aprendizagem profunda.

Abstract

Image retargeting involves adjusting input images to arbitrary dimensions (rows and columns) while preserving regions of interest. Evaluating image quality under varying aspect ratios is particularly challenging as it requires content matching and semantic content analysis. In this study, we introduce two objective quality assessment algorithms for image retargeting. The first one is an autoencoder-based approach named AutoEncoder Information MaP Similarity (AEIMPS). It uses the retargeting ratio, the latent space and the reconstructed image in its pipeline. The second one, inspired by the first, is the Encoding MaP Image Retargeting Quality (EMPIRQ). It is a no-reference approach that assesses quality in image retargeting through regressor architectures that take encodings computed by autoencoders as input. We evaluated our proposal on a widely recognized state-of-the-art dataset, where human viewers provided personal opinions on perceptual quality. Additionally, we proposed five baseline models to analyze bias towards results. EMPIRQ achieved outstanding results: LLC of 0.98, SRCC of 0.98, RMSE of 3.06, and OR of 0.01, surpassing all baseline models and state-of-the-art algorithms under comparison.

Keywords: autoencoder. image retargeting quality assessment. deep learning.

List of Figures

Figure 1 – Example of image retargeting: the original image (a) and the retargeted results from manual cropping (b), scaling (c), and the Shift-map operator, proposed by Pritch, Kav-Venaki e Peleg (2009) (d). As evident, in (b), content is discarded through cropping, whereas in (c), scaling introduces uniform distortion in each pixel. Finally, in (d), the Shift-map operator yields the most favorable outcome.	35
Figure 3 – Examples of retargeted images from RetargetMe dataset.	36
Figure 4 – Artificial neuron.	44
Figure 5 – Feedforward Network.	44
Figure 6 – 2D convolution operation.	46
Figure 7 – Example of a CNN.	47
Figure 8 – A convolution layer with two kernels, no padding and a stride value of 1.	47
Figure 9 – Autoencoder architecture.	48
Figure 10 – Autoencoder classification in terms of the latent space dimensionality.	49
Figure 12 – Residual Block.	50
Figure 13 – Examples of famous ANNs architectures.	51
Figure 15 – Cross-validation methodologies.	51
Figure 16 – The double-descent risk curve.	53
Figure 17 – AEIMPS flowchart.	58
Figure 18 – EMPIRQ flowchart.	59
Figure 19 – Resizing with padding.	62
Figure 21 – LLC metric for all encoding embeddings $\times$ regressor combinations grouped by cross validation methodology.	65

List of Tables

Table 1 – IRQAs comparison by statistical scores.	41
Table 2 – Results of each encoding embeddings with regressor combination for EMPIRQ.	64
Table 3 – IRQAs comparison by statistical scores with proposals and baselines. . .	66
Table 4 – Fusion strategies results.	80

List of abbreviations and acronyms

AEIMPS	AutoEncoder Information MaP Similarity.
ANN	Artificial Neural Network.
CMAE	Complement of Mean Absolute Error.
CMY	Cyan, Magenta, Yellow.
CMYK	Cyan, Magenta, Yellow, Black.
CNN	Convolutional Neural Network.
CONV	Convolution.
CR	Cropping.
DENSE	Dense Layer.
DL	Deep Learning.
DMOS	Difference Mean Opinion Score.
E2E	End-to-End.
EH	Edge Histogram.
EMD	Earth Mover's Distance.
EMPIRQ	Encoding MaP Image Retargeting Quality.
ENER	Energy-based deformation.
FR	Full Reference.
GLCM	Gray Level Co-occurrence Matriz.
H50	Holdout (80/20).
H80	Holdout (50/50).
INSEM	INstance SEMantics.
IQA	Image Quality Algorithm.
IRQA	Image Retargeting Quality Assessment.

KFO	K-Fold.
LCC	Linear Correlation Coefficient.
LOO	Leave-One-Out.
LOG	Leave-One-Group-Out.
ML	Machine Learning.
MLF	Multiple-Level Feature.
MLP	Multilayer Perceptron.
MOS	Mean Opinion Score.
MULT	Multi-operator process.
NR	No Reference.
OR	Outlier Ratio.
POOL	Pooling.
PSRN	Peak Signal-to-Noise-Ratio.
RGB	Red, Green, Blue.
RMSE	Root Mean Square Prediction Error.
RR	Reduced Reference.
SCL	Scaling.
SCSC	Optimized seam carving and scale.
SCST	Scale and stretch.
SEAM	Seam Carving.
SGD	Stochastic Gradient Descent.
SHIF	Shift-map editing.
SIFT	Scale-invariant feature transform.
SRCC	Spearman Rank-Order Correlation Coefficient.
SSIM	Structural Similarity Index Measure.
STVI	Streaming video.

VAE	Variational Autoencoder.
VAL	Validation.
VQEG	Video Quality Experts Group.
WARP	Non-homogeneous retargeting.

List of symbols

$\mathbf{I}$	original image.
$\mathbf{J}$	retargeted image.
m	original image height.
n	original image width.
m'	retargeted image height.
n'	retargeted image width.
β_i	regression model parameter.
$f(\cdot)$	five-parameter logistic regression function.
ρ	retargeting ratio.
$e(\cdot)$	encoder function.
$d(\cdot)$	decoder function.
σ	activation function.
x	input vector.
n_x	input vector size.
w_i	i -th weight connection.
b	bias term.
$\mathbf{T}$	a tensor.
$\mathbf{Z}$	output from a convolution.
$\mathbf{K}$	a kernel.
m_k	kernel height.
n_k	kernel width.
k	number of folds.
$\mathcal{H}$	hyphothesis.

Contents

1	INTRODUCTION	29
1.1	Motivation	29
1.2	Objectives	30
1.3	Publications	31
1.4	Document organization	31
2	IMAGE RETARGETING QUALITY ASSESSMENT	33
2.1	Image retargeting	33
2.2	Evaluating the image retargeting quality	35
2.3	Evaluating IRQA	40
3	MACHINE LEARNING	43
3.1	Artificial Neural Networks	43
3.1.1	Convolutional Networks	45
3.1.2	Autoencoders	47
3.2	ANN architectures	49
3.3	Training and evaluation steps	50
3.4	Machine Learning and Deep Learning	53
4	PROPOSALS	55
4.1	Baselines	55
4.2	AEIMPS	56
4.3	EMPIRQ	58
5	EXPERIMENTS AND RESULTS	61
5.1	Experiments	61
5.2	Results and discussion	63
5.2.1	Encoding embeddings with regressor combinations analysis for EMPIRQ . .	63
5.2.2	Baseline analysis	64
5.2.3	CUHK benchmark analysis	65
6	CONCLUSION	69
	BIBLIOGRAPHY	71

APPENDIX	77
APPENDIX A – FUSION COMBINATION STRATEGIES FROM AEIMPS	79

1 Introduction

Many applications utilize various forms of data, including text, audio, and images. Images, in particular, are closely tied to essential patterns for application integration, such as file size, format, and dimensions. As a result, there is a growing demand for operators that can adapt images to ensure compatibility and optimal visualization within these applications. One such crucial adjustment is image retargeting, which involves modifying the dimensions of image data to meet specific height and width requirements.

The image retargeting problem can be stated as follows. Let $\mathbf{I}$ be an input image of size $m \times n$, where m is the number of rows and n is the number of columns. Similarly, let $\mathbf{J}$ be an output image of size $m' \times n'$, where $m' = m$ and $n' < n$, for a width reduction scenario. The goal is then to produce a new image $\mathbf{J}$ which will be a good representative of the original image $\mathbf{I}$. Roughly speaking, image retargeting is in general applied to (i) preserve the input image content, (ii) preserve the input image structure, as well as to (iii) achieve a resulting artifact-free image (SHAMIR; SORKINE, 2009).

One could achieve this task by either cropping or scaling the image proportionally. However, such methods can lead to a poor representation of the original image by removing or distorting essential objects. This issue arises because these methods do not account for important aspects of the image. Content-aware image retargeting operators address this problem by focusing on regions of interest within the original image (SHAMIR; SORKINE, 2009; VAQUERO et al., 2010). As a result, they generally produce more aesthetically pleasing results compared to conventional operators. The basic idea behind content-aware image retargeting is to identify low-interest regions and then either expand or reduce them to create a retargeted image with minimal noticeable changes to the observer.

1.1 Motivation

In order to select the most suitable retargeting operator, an evaluation method is needed to score their performances. Designing an objective quality metric for image retargeting that compares content across varying aspect ratios is significantly more challenging, as it requires semantic image analysis and content matching (RUBINSTEIN et al., 2010). Similarly, popular Image Quality Assessment (IQA) metrics, such as the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index Measure (SSIM) (WANG et al., 2004), or even the Mean Absolute Error, cannot be applied directly, as they require the reference and retargeted images to have the same dimensions.

Three very interesting studies, one conducted in 2010 (RUBINSTEIN et al., 2010),

2012 (MA et al., 2012), and another in 2015 (MA et al., 2015) found fascinating outcomes related to the subjective component over the image retargeted quality assessment (IRQA). The first study found that humans generally agree with each other on the quality of retargeted images, and some retargeting algorithms are consistently more favorable than others. However, correlations yielded by IRQAs with subjective evaluations were not always consistent (RUBINSTEIN et al., 2010). The subsequent study revealed that human subjects exhibit high sensitivity to distortions in faces and geometric structures, while being more tolerant of distortions in natural scenery, particularly in texture regions (MA et al., 2012). The latter study conducted in (MA et al., 2015) discussed how to design an effective IRQA by considering the reference image content, retargeting scale, the shape distortion and content information loss, the Human Vision System properties, among other descriptors. In all studies, the authors highlighted that the assessed IRQA performances are still unsatisfactory by the same statistical correlation between scores, indicating that there is still room for improvements.

Moreover, Deep Learning (DL) techniques have also been explored in IRQAs, as proposed by Yan et al. (2017) and Fu et al. (2018). One such technique is to employ autoencoders, which are artificial neural networks that reconstruct input data by computing encoded data as an intermediate step. Numerous enhancement techniques for autoencoders involve processing within the hidden layers to make the encoded data as different from the input data as possible. When the reconstruction error between the input and output is minimal, it indicates that the hidden layer's encoding has captured meaningful features learned by the model (LI; PEI; LI, 2023).

From this perspective, this dissertation explores the following research question:
Could such encoded data (embeddings) from AEs be effectively employed as features to design a proper IRQA?

1.2 Objectives

This work tackles the problem of image retargeting quality assessment by proposing an IRQA based on deep embeddings and different deep architectures. Adopting deep embeddings is the manner we found to take into account the loss of relevant content, as well as, the introduction of visual artifacts in retargeting results. Specifically, this work has the following objectives:

- a) To extract features from deep autoencoders in order to predict the quality of retargeted images as perceived by users;
- b) To evaluate these deep embeddings using regressor architectures that incorporate the following mechanisms: residual connections, attention mechanisms, and simple convolution operators;

- c) To provide an analysis using baseline models to discuss the nature and performance of various state-of-the-art IRQAs.

We highlight our contributions as follows:

- a) Five baseline models for individual comparison of IRQA: Expert, Amateur, ρ , Permutation, and Chance;
- b) Two autoencoder-based IRQA methods that apply deep embeddings: AEIMPS and EMPIRQ.

1.3 Publications

During this master’s project, the following scientific papers were developed:

- a) **Levi C. Carvalho** e Saulo A. F. Oliveira. “Aeimps: Deep autoencoder for image retargeting quality assessment”. XXXV Conference on Graphics, Patterns and Images, SIBGRAPI. 2022.
- b) **Levi C. Carvalho** e Saulo A. F. Oliveira. “Exploring-exploiting in Deep Embeddings for Image Quality Retargeting Assessment”. To be submitted.

1.4 Document organization

This document is composed of six chapters that cover various concepts related to IRQA methods, qualitative evaluation of autoencoder-based IRQA results, and machine and deep learning techniques. More specifically, Chapter 2 describes the image retargeting quality assessment (IRQA) problem, highlights various IRQA methods, and provides an overview of how to qualitatively evaluate their results, including the evaluation methods used in the experiments. Chapter 3 presents the general aspects of machine learning that are fundamental for the development of this dissertation and for understanding our proposed methods. Chapter 4 introduces two autoencoder-based proposals for image retargeting quality assessment: AutoEncoder Information MaP Similarity (AEIMPS) and Encoding MaP Image Retargeting Quality (EMPIRQ), along with the baseline models used for comparison. Chapter 5 details the experiments conducted to evaluate the proposed methods. Finally, Chapter 6 provides the final considerations, comments, and analysis of the results obtained in this dissertation.

2 Image Retargeting Quality Assessment

In this chapter, we present key concepts associated with image retargeting, elucidating their definition and providing illustrative examples. Additionally, we introduce the main image retargeting quality assessment datasets, essential resources to conduct comprehensive experiments. Moreover, we conclude this chapter by delving into the methodologies for assessing retargeting results, elucidating various scoring mechanisms alongside associated evaluation criteria.

2.1 Image retargeting

The image retargeting problem is formulated as follows: Let $\mathbf{I}$ be an original image with dimensions $m \times n$, where m represents the height and n the width. The objective is to resize (or to retarget) such an image to dimensions $m' \times n'$ using a designated operator, resulting in the retargeted image $\mathbf{J}$. Additionally, it is important to note that m or n may or may not be equal to m' or n' . Ideally, the retargeted image should preserve its content. As discussed in Shamir e Sorkine (2009), image retargeting is generally applied to:

- a) Preserve the original image content;
- b) Preserve the original image structure;
- c) Achieve a resulting artifact-free.

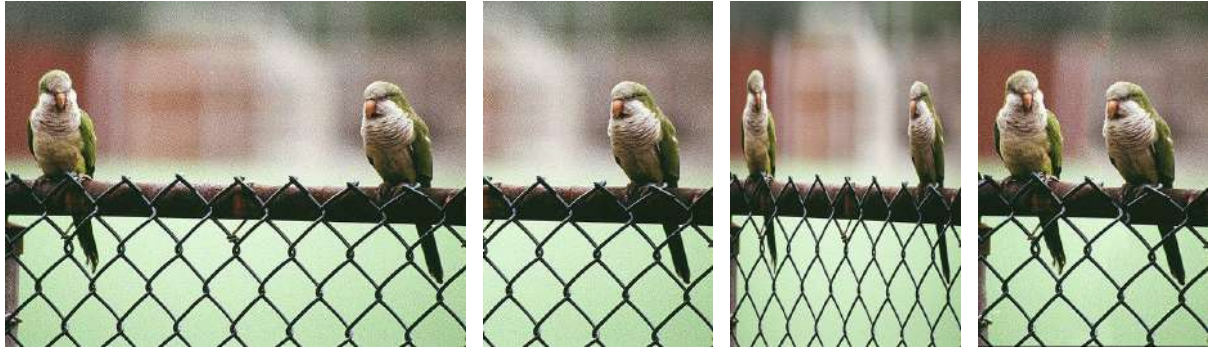
In the following, we cite ten retargeting operators from the State-of-the-art alongside their short description.

- a) **Cropping (CR)** entails a selecting window that matches the desired dimensions. Such an operator strategically isolate and retain key features (pertinent content) within the resized image, ensuring capturing significant elements;
- b) **Scaling (SCL)** involves resizing the original image to a specified dimension through the implementation of an interpolation algorithm, such as bilinear or bicubic interpolation;
- c) **Seam Carving (SEAM)** (AVIDAN; SHAMIR, 2007): removes or adds seams from the original image. A seam is essentially a sequence of pixels that starts at the top-left and extends to the bottom-right of the image, with each successive pixel belonging to the 8-neighborhood of the previous one. To reduce the size of the original image, such seams should ideally contain minimal relevant information from the original image;

- d) **Optimized seam carving and scale (SCSC)** (DONG et al., 2009) applies seam carving and scaling to retarget an image. Such a combination has the ability to preserve the global visual impact and certain local structures of the original image, particularly when the resulting resolution is significantly lower than the original resolution;
- e) **Non-homogeneous retargeting (WARP)** (WOLF; GUTTMANN; COHEN-OR, 2007) optimizes a warping function to calculate the resized image. A per-pixel importance measure drives this operator. Such a measure is the combination of a local saliency based on gradients, a face detector, and a high-end motion detector;
- f) **Scale and stretch (SCST)** (WANG et al., 2008) optimizes a goal function by uniformly adjusting the salient areas to maintain the shape details. The saliency map is determined by integrating the gradient magnitude with the saliency map identified by Itti, Koch e Niebur (1998);
- g) **Shift-map editing (SHIF)** (PRITCH; KAV-VENAKI; PELEG, 2009) applies a graph cutting algorithm to perform object removal through the shift-map usage. This shift-map is computed based on constraints and a smoothness term, which incorporates color differences and gradient information;
- h) **Multi-operator process (MULT)** (RUBINSTEIN; SHAMIR; AVIDAN, 2009) combines three retargeting operators: Seam Carving, scaling and cropping. It utilizes an image similarity measure known as Bi-Directional Warping with dynamic programming to determine the optimal usage of such operators;
- i) **Energy-based deformation (ENER)** (KARNI; FREEDMAN; GOTSMAN, 2009) minimizes a deformation energy to retarget an image. Such a deformation energy indicates acceptable deformations types through the allowed transformations sets;
- j) **Streaming video (STVI)** (KRÄHENBÜHL et al., 2009) employs a non-uniform and pixel-accurate warping technique that utilizes automatic features, including video saliency, pixel-resolution edge preservation, and scene cut detection.

In Figure 1, one can see a retargeting example employing three operators, namely, cropping, scaling, and Shift-map (PRITCH; KAV-VENAKI; PELEG, 2009). We resized the original image to retain 50% of its width, while its height remained unchanged. As one can see, the Shift-map operator was the only capable of preserving the original content while maintained the two birds without significant distortions due to its content-aware nature.

Figure 1 – Example of image retargeting: the original image (a) and the retargeted results from manual cropping (b), scaling (c), and the Shift-map operator, proposed by Pritch, Kav-Venaki e Peleg (2009) (d). As evident, in (b), content is discarded through cropping, whereas in (c), scaling introduces uniform distortion in each pixel. Finally, in (d), the Shift-map operator yields the most favorable outcome.



(a) Original image.

(b) Cropping.

(c) Scaling.

(d) Shift-map.

Source: Carvalho e Oliveira (2022)

2.2 Evaluating the image retargeting quality

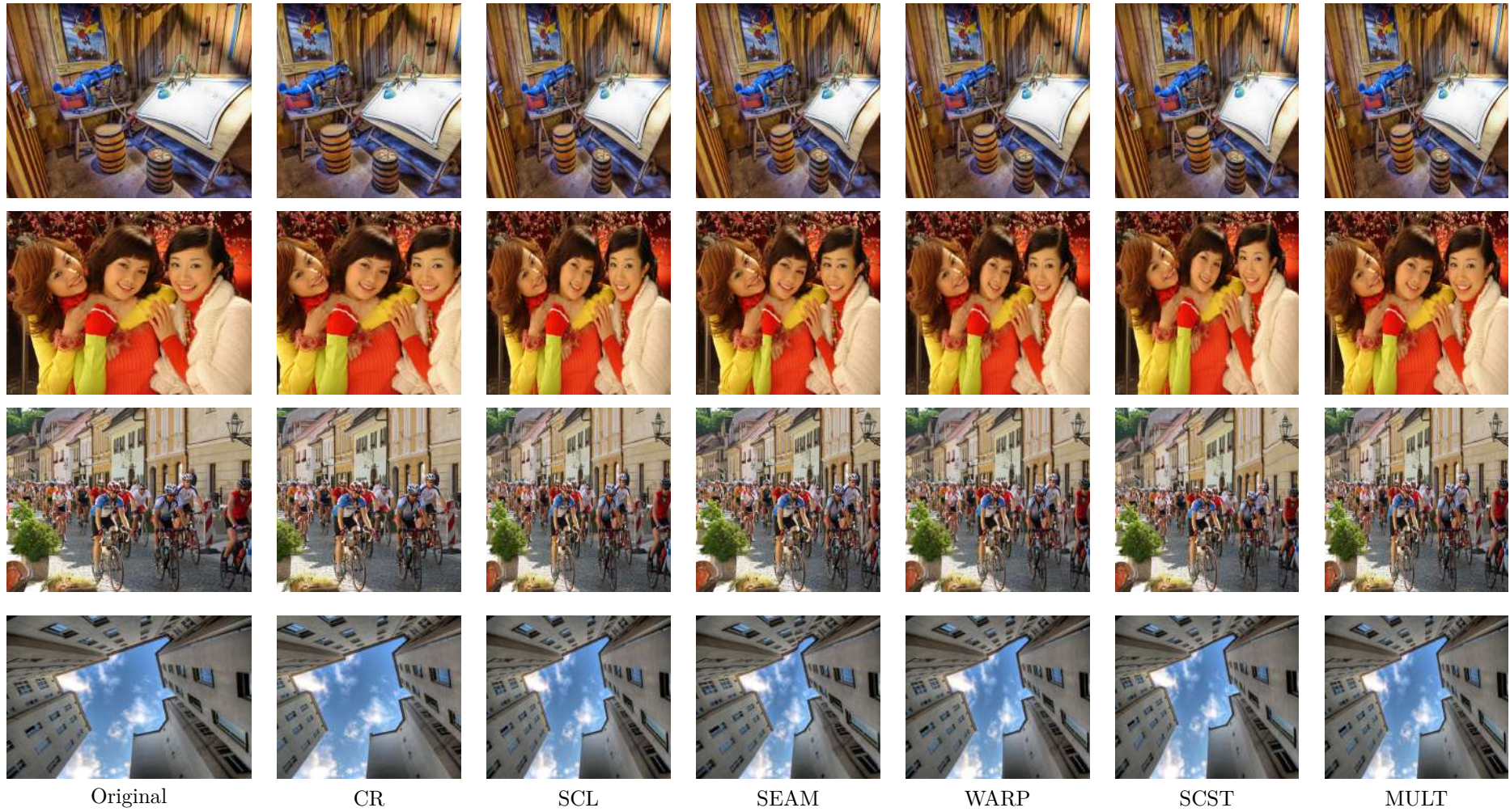
As we discussed earlier, there are many retargeting algorithms that one can employ to resize an image. The most suitable choice depends on the application and its fault tolerance. For example, an image with anomalies on a social network is less harmful than on a medical application. Such anomalies can be distortions or artifacts.

Figure 3 illustrates retargeted image examples by some operators. It becomes evident that certain algorithms exhibit superior performance compared to others. We can observe that some faces exhibit anomalies depending on the retargeting algorithm applied. Additionally, geometric shapes in the images, such as the space between buildings in the last images, show noticeable distortions. Some algorithms, like cropping, do not introduce anomalies or distortions but result in the loss of information from the original image.

To make informed decisions about the most suitable algorithm for a given context, it is crucial to employ objective metrics that systematically rank such algorithms based on their performance. Such an approach ensures a more comprehensive evaluation and facilitates the understanding of each algorithm's strengths and weaknesses.

The evaluation of the distortion introduced by adding noise to an image, for instance, is relatively straightforward. This simplicity arises from the fact that the original and distorted images share the same dimensions. Consequently, one can apply popular Image Quality Algorithms (IQAs) that performs a pixel-by-pixel evaluation between both images, such as the Peak Signal-to-Noise-Ratio (PSNR), the Structural Similarity Index Measure (SSIM) (WANG et al., 2004), or even the Mean Absolute Error. However, assessing the quality of a retargeted image cannot be conducted at the same pixel level. Such behavior is due to the incompatible dimensions of the original and resized image.

Figure 3 – Examples of retargeted images from RetargetMe dataset.



Source: Author.

Image retargeting quality assessment (IRQA) is the task of evaluate the quality of these retargeted images. Instead of a pixel by pixel strategy, IRQA performs a semantic evaluation on the problem. It seek to explore image elements by capturing relevant features from the images. Then, an algorithm uses such features to objectively compute the opinion measure.

As one can see, IRQA involves two primary objects to address its problem: the original image and the retargeted image. Therefore, IRQA metrics can be classified into one of three categories of objective image quality assessment methods: No Reference (NR), Reduced Reference (RR), and Full Reference (FR). NR addresses the IRQA task in a blind manner, without employing the original image. Conversely, FR utilizes all the available information from the IRQA task, including both the retargeted and original images. Finally, RR employs some representation of the original image, such as automatic or manual features.

The benchmark datasets from the IRQA problem usually have some numerical value for human opinion scores. Such values represent an subjective evaluation of a particular retargeted image or a ranking between retargeted images by a pairwise comparison strategy. RetargetMe (RUBINSTEIN et al., 2010) and NRID (HSU et al., 2013) are popular datasets for this pairwise comparison. While CUHK dataset (MA et al., 2012) is a common choice for the subjective evaluation strategy. Two popular ways of evaluating such retargeted image are the Mean Opinion Score (MOS) and Difference Mean Opinion Score (DMOS). A MOS, serving as a subjective metric, is assigned to each pair of original and retargeted images to assess the efficacy of the retargeting operator. The MOS value for a given pair is calculated as the average of retargeting quality scores provided by individuals, who may or may not possess expertise in the field. The DMOS is the difference between "reference" and "processed" MOS in a full reference testing.

In the following, we cite sixteen IRQA from the State-of-the-art alongside. We employ such metrics to compare our results. As some works did not define names for their proposed IRQA, we named them based on its main author and year.

- a) **Yan2017** (YAN et al., 2017) employs a RBF network to predict the quality score of the retargeted image. The network input is ten features computed from the retargeted image, including a saliency-weighted, SIFT-directed complex wavelet structural similarity index;
- b) **INSEM** (LI et al., 2021) proposes an image retargeting quality metric with INstance SEMantics (INSEM). It performs instance segmentation and extracts a set of semantic features. Then, such an information is passed to the semantic-based self-adaptive pooling. The final step is a combination between instance features and a global feature to generate the overall quality score;

- c) **RN-IRQA** (NIU et al., 2021) is an IRQA index based on registration confidence measurement and noticeability-based pooling. The RCM alleviates the negative influence of inaccurate registration on fidelity measurement. The NBP strategy improves the performance of IRQA for inter-set comparison;
- d) **Li2019** (LI; GUO; JIN, 2019) proposes a IRQA framework which applies foreground and global measurements. Foreground measurement includes the high-level semantic similarity feature and low-level size ratio feature. While the global measurement includes an improved aspect ratio similarity feature and edge group similarity feature. A backward registration is used to estimate the pixel-level correspondence between the original and retargeted images. Then, if an obvious foreground object is present in the image, it applies both foreground and global measurement. For others, only global measurement is conducted;
- e) **MLF** (ZHANG et al., 2018) is a Multiple-Level Feature (MLF)-based quality measure. First, it estimates pixel-level correspondences by the backward registration method. Then, it generates multiple-level features to capture changes due to retargeting. Aspect Ratio Similarity measures local information loss and distortion, Edge Group Similarity focuses on structural changes, and Face Block Similarity targets facial deformation. Finally, a fusion model combines these features to predict the quality of the retargeted image;
- f) **HCnDL** (FU et al., 2018) is inspired by deep convolutional neural networks and combines hand-crafted and deep-learned features for IRQA. Traditional hand-crafted features fail to capture structure distortion and content loss from retargeting. To address this, the metric uses similarity transformation to relate source and retargeted images, employs a pyramidal model for multi-scale structure distortion, and measures area changes in grids for content loss. Additionally, a CNN extracts deep-learned features, including texture and semantic features, enhancing IRQA prediction accuracy;
- g) **Karimi2017** (KARIMI et al., 2018) employs a pyramidal pooling framework to evaluate both global and local geometric distortions in retargeted images. This approach divides the image into cells at different pyramid levels, calculates local deformation averages, and uses these features to train a machine learning algorithm for objective quality scoring. It incorporates aspect ratio and area as key features to capture significant geometric distortions and adjusts their weights based on saliency;
- h) **ARS** (ZHANG et al., 2016) uses geometric change estimation, formulated as a Backward Registration problem with Markov Random Field, to clarify the relationship between the original and retargeted images. Based on this geometric change, it computes an Aspect Ratio Similarity metric that evaluates retargeted

image quality by analyzing local block changes through a visual importance pooling strategy;

- i) **HDPM** (LIN et al., 2015) performs the IRQA task by evaluating image quality through three aspects: local similarity, content information loss, and structural distortion. The method generates SIFT descriptors for local similarity, uses unmatched descriptors to assess content loss, and applies GLCM features to measure structural distortion. These aspects are then combined to estimate the overall image quality;
- j) **PGDIL** (HSU et al., 2013) is an objective quality metric for assessing retargeted images by evaluating perceptual distortion and information loss. The metric measures local geometric distortion using SIFT flow estimation and incorporates a visual saliency map to reflect human perception of these distortions. Additionally, it quantifies information loss based on the saliency map to assess the extent of important information lost during retargeting.;
- k) **Liu2011** (LIU et al., 2011) proposes an objective assessment metric for retargeted images that uses a top-down approach. The metric organizes image features from global to local viewpoints. It employs a scale-space matching method to extract global geometric structures and establish local pixel correspondences by traversing from coarse to fine scales. The assessment combines both global structures and local correspondences to evaluate image quality;
- l) **EH** (MANJUNATH et al., 2001) proposes the Edge Histogram (EH) descriptor that captures an image’s edge distribution by dividing it into 4×4 sub-images and analyzing five edge orientations: vertical, horizontal, two diagonals, and isotropic. Each sub-image generates a 5-bin histogram, resulting in a fixed-length descriptor of 80 values. The image intensity component is used for edge extraction, and L1 distance is typically used to measure distance between two EH descriptors;
- m) **SIFT-flow** (LIU et al., 2008) proposes a method for image registration and correspondence. It builds on the Scale-Invariant Feature Transform (SIFT) algorithm, which detects and describes local features in images. A SIFT descriptor is extracted at each pixel to characterize local image structures and encode contextual information. Then, SIFT-flow matches the SIFT descriptors between two images through a flow estimation algorithm;
- n) **BDS** (SIMAKOV et al., 2008) employs a patch-based bidirectional similarity measure between visual data of different sizes. Two images are considered visually similar if as many as possible patches of one (at multiple scales) are contained in the other, and vice versa;

- o) **EMD** (PELE; WERMAN, 2009) is a fast algorithm for the computation of the EMD (Earth Mover's Distance) with thresholded ground distance. Conceptually, it represents the minimum cost of transforming one distribution into another by moving "distribution mass" in the most efficient way;
- p) **IR-SSIM** (FANG et al., 2014) is a SSIM-based algorithm to compute an objective quality assessment algorithm for image retargeting. The score is computed by weighting the SSIM map between the reference and retargeted images using the overall saliency map, which is a combination between bottom-up and top-down saliency maps.

2.3 Evaluating IRQA

The IRQA introduced in this study, along with those from other references, aims to forecast a numerical representation indicating the effectiveness of image retargeting, whether or not the original image is being employed. As the objective is for such a numerical output to closely align with a subjective score a human would assign, a comprehensive ranking of IRQAs using some scores is necessary. This ranking allows for close alignment with human perception of retargeting quality.

The Video Quality Experts Group (VQEG) HDTV employs an approach that proposes four statistical scores, facilitating a comparative analysis of the predicted MOS values generated by the assessed algorithms. These scores are the Linear Correlation Coefficient (LCC), the Spearman Rank-Order Correlation Coefficient (SRCC), the Root Mean Square Prediction Error (RMSE), and the Outlier Ratio (OR). Notably, this methodology aligns with the practices outlined in previous studies such as Ma et al. (2012), Zhang et al. (2016), and Liu et al. (2011). In this context, higher LCC and SRCC coefficients indicate stronger agreement, while lower RMSE and OR values reflect fewer discrepancies between the original and predicted MOS. The LCC, SRCC, and OR range between 0 and 1, while RMSE is at least 0. This rigorous evaluation strategy ensures a comprehensive and objective assessment of algorithmic performance in predicting retargeted image quality.

To compute these scores and obtain a precise evaluation of the IRQA, it becomes necessary to transform the set of outputs coming from the algorithm. In this study, a five-parameter logistic function maps each score. The Equation 2.1 illustrates such a function, where $\{\beta_i\}_{i=1}^5$ represents the regression model parameters. The minimization of the sum of squared differences between IRQA outputs and real MOS determines such parameters that latter are passed to compute statistical scores as the ones seen in Table 1. Later, in Section 2.2, we present them.

$$f(x) = \beta_1 \left(\frac{1}{2} - \frac{1}{\exp(\beta_2(x - \beta_3))} \right) + \beta_4 x + \beta_5 \quad (2.1)$$

Table 1 – IRQAs comparison by statistical scores.

IRQA	LLC	SRCC	RMSE	OR
Yan2017 (YAN et al., 2017)	0.921	0.912	3.201	0.011
INSEM (LI et al., 2021)	0.798	0.748	7.905	0.023
RN-IRQA (NIU et al., 2021)	0.792	0.761	8.234	0.040
Li2019 (LI; GUO; JIN, 2019)	0.774	0.770	8.542	0.070
MLF (ZHANG et al., 2018)	0.757	0.738	8.525	0.029
HCnDL (FU et al., 2018)	0.717	0.684	9.135	0.021
Karimi2017 (KARIMI et al., 2018)	0.716	0.709	9.639	0.059
ARS (ZHANG et al., 2016)	0.683	0.669	9.855	0.070
HDPM (LIN et al., 2015)	0.671	0.690	-	-
PGDIL (HSU et al., 2013)	0.540	0.540	11.361	0.152
Liu2011 (LIU et al., 2011)	0.437	0.466	12.141	0.152
EH (MANJUNATH et al., 2001)	0.342	0.328	12.686	0.204
SIFT-flow (LIU et al., 2008)	0.314	0.290	12.817	0.146
BDS (SIMAKOV et al., 2008)	0.289	0.288	12.922	0.216
EMD (PELE; WERMAN, 2009)	0.276	0.290	12.977	0.169
IR-SSIM (FANG et al., 2014)	0.230	0.240	13.140	0.176

Source: (ASHEGHI et al., 2022)

3 Machine Learning

As our proposals performs automatic feature extraction via embeddings from autoencoders, this chapter delves into a comprehensive discussion on this subject, elucidating crucial concepts in machine learning, deep learning, and the application of autoencoders. We emphasize that most of the theoretical foundations presented in this chapter can be found in the works of Goodfellow, Bengio e Courville (2016), LeCun, Bengio e Hinton (2015) and Russell e Norvig (2020).

Machine Learning (ML) is the capability to acquire knowledge by extracting patterns from raw data. According to Russell e Norvig (2020), ML has two main advantages over hard-coded solutions: there is no need to anticipate all possible future situations and the ML model finds the solution itself. ML often solves tasks that are trivial for human beings, but hard for machines, such as face recognition and speech transcription. On the other hand, when it comes to tasks that are difficult for human beings, ML can also solve them and can even achieve a superior result.

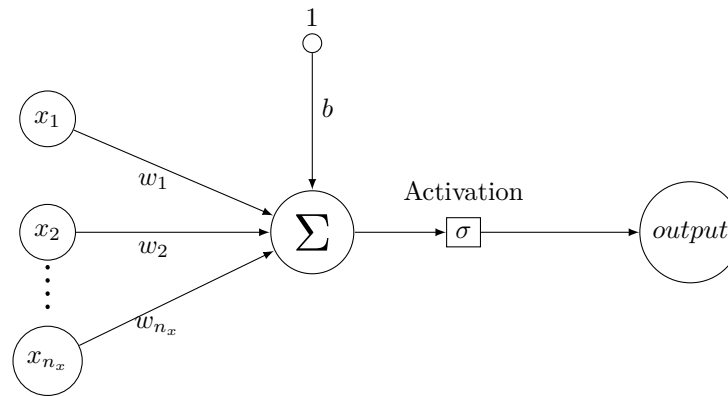
3.1 Artificial Neural Networks

Artificial Neural Networks (ANNs) are circuits that learn patterns from the data inspired in the biological neural network. They have as one of its main influences the work of McCulloch e Pitts (1943), that tried to model networks of neurons in the brain through computational circuits. The smallest computation node is the artificial neuron, also called unit. Nowadays, modern ANNs still applies its proposed design, which is presented in Figure 4. The neuron computes a weighted sum of the inputs, besides a bias term, and then applies a nonlinear activation function σ , as presented in equation 3.1. The neuron parameters, also called weights, are learned in the training step and are represented by variables w_i and b .

$$output = \sigma \left(\sum_{i=1}^{n_x} w_i x_i + b \right). \quad (3.1)$$

One can build ANNs using different types of architectures, but one of the simplest is the feedforward network. In such architecture, we build the network by organizing the neurons into layers, as presented in Figure 5. The first layer is known as input layer and receives the sample data, while the last layer is the output layer and gives the output for the problem. The intermediate layers are called hidden layers, which the neurons of each one learns a similar function in terms of weight and bias as the one in equation 3.1. Note that a function of a neuron couples multiples functions of previous layers neurons. Such

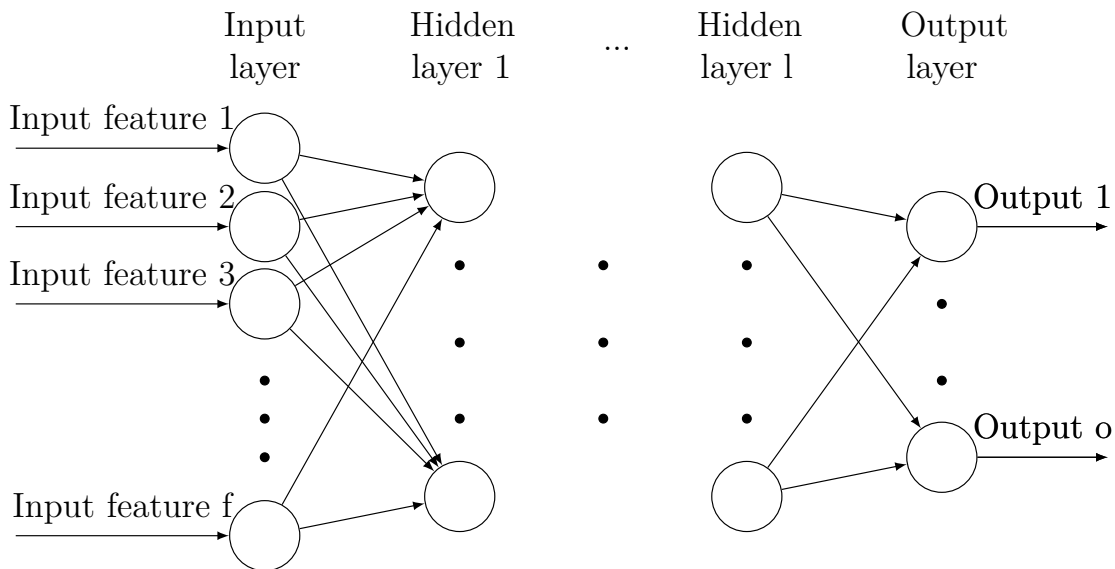
Figure 4 – Artificial neuron.



Source: Author.

behavior enables a network learning of complex functions, which is necessary given that real-world problems are usually mapped through very complex functions. The feedforward network gets its name because the data flow for inference runs in only one direction: from the input layer to the output layer.

Figure 5 – Feedforward Network.



Source: Author.

We can interpret the internal computation of a neuron as a function, and we know that any composition of linear functions could be represented as a single linear function (RUSSELL; NORVIG, 2020). A single neuron could learn by itself this single linear function, resulting in unnecessary use of more neurons in other layers. Therefore, we need to use non-linear activation functions to enable the network to learn more complex functions.

The ANNs training step consists of updating the parameters values of each neuron of hidden and output layers. There is a loss function that guides this update process and it must be convex in order to have convergence guarantee (GOODFELLOW; BENGIO; COURVILLE, 2016). Chose a loss function depends on the problem to solve. In order to update the output layer weights, we perform the gradient descent algorithm: it computes the loss function gradients concerning the weights and modifies the weights in the opposite gradient direction to minimize the loss. We adjust the hidden layer parameters by applying a slight different approach: back-propagation. Such method propagates the output layer error to previous layers by using chain rule from calculus. When we perform this parameters update for each sample, this whole process is known as Stochastic Gradient Descent (SGD). We can apply some extensions and variants to slight modify this process in order to perform a better learning, such as performed by Adam optimizer (KINGMA; BA, 2014).

Next, we discuss some elements, operations and properties present in two special types of ANNs: Convolutional Networks and Autoencoders.

3.1.1 Convolutional Networks

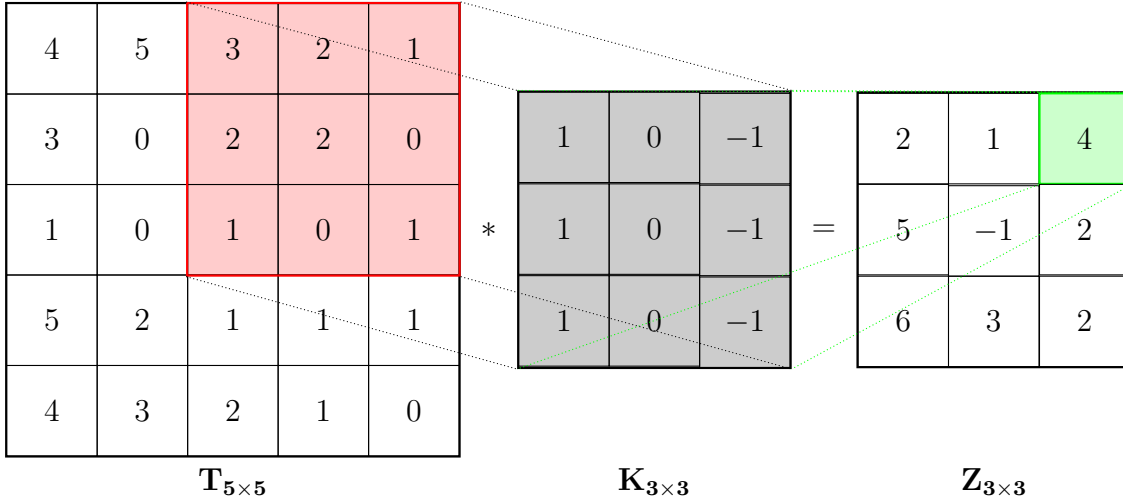
Convolutional networks (LECUN et al., 1989), or CNNs, are a specialized kind of ANN for processing data that has a known grid-like topology. This data type has a important property that we must pay attention: spacial data. The resulting from this training could be computed whether we trained with unperturbed images or with images all of whose pixels had been randomly permuted (RUSSELL; NORVIG, 2020). CNNs solve this problem by using the mathematical operation that characterizes them: convolutions.

A convolution is a mathematical operation that is denoted by $*$ symbol. Such operation slides horizontally and vertically a pattern of weights $\mathbf{K}$, a $m_k \times n_k$ grid known as kernel, over spatially equal regions of a tensor t , resulting in the output $\mathbf{Z}$. For each region, we perform a dot product between the kernel and the corresponding region. The resulting value is stored in its corresponding position in the output $\mathbf{Z}$. The convolution value at point (a, b) is given by equation 3.2. We demonstrate such operation with an example in Figure 6. Note that, for image applications, $\mathbf{T}$ and $\mathbf{K}$ are usually 3D tensors and must have the same number of channels, summing the convolution result along each corresponding channel and resulting in a matrix.

$$\mathbf{Z}[a, b] = (\mathbf{T} * \mathbf{K})[a, b] = \sum_{i=-m_k}^{m_k} \sum_{j=-n_k}^{n_k} \mathbf{T}[a-i, b-j] \mathbf{K}[i, j] \quad (3.2)$$

Additionally, we can configure the kernel sliding by two hyperparameters: stride and padding. Note that we slide horizontally and vertically the kernel one position at a time in Figure 6. We can change this shift size by the stride hyperparameter. In CNNs, an important motivation for using strides greater than one is data reduction. Moreover, we

Figure 6 – 2D convolution operation.



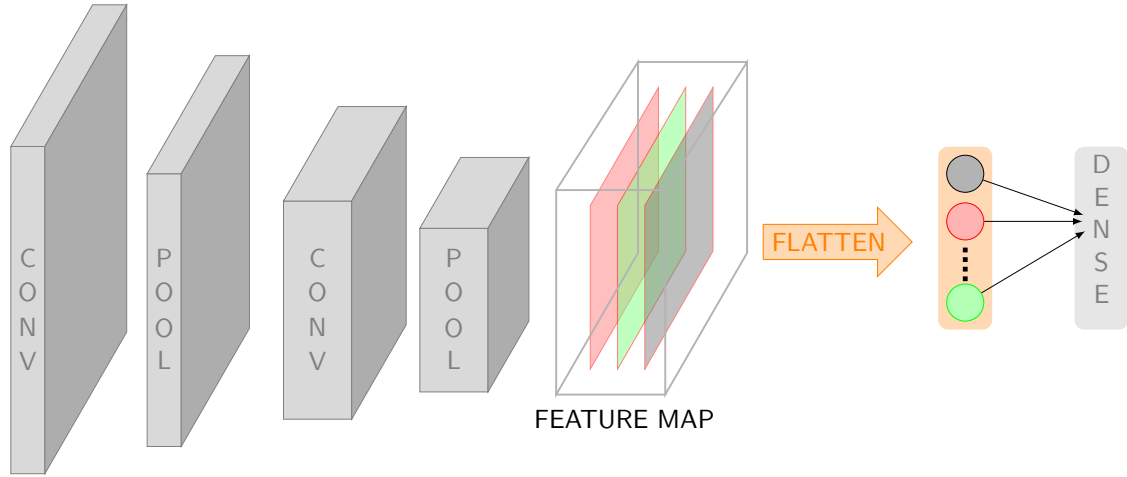
Source: Author.

can increase the image border a certain number of times using the padding hyperparameter. We can apply many approaches to execute this action, but a popular choice is to just pad it with zero values.

A convolutional network is one that applies the convolution operation at some point in its architecture. We can see a brief example at Figure 7. Note that it is composed by convolutional and pooling layers. As the name suggests, convolutional layers are responsible for applying the convolution operation, alongside a point-by-point addition of the convolution result with a bias and a non-linear activation function. The kernel values and bias are parameters which are learned in the training step. It applies a certain amount of kernels which results in in the same amount of matrices that are stacked at the final tensor output, which is sometimes referred to as feature map. We can see an example of how a convolution layer works at Figure 8 with two kernels, no padding and a stride value of 1.

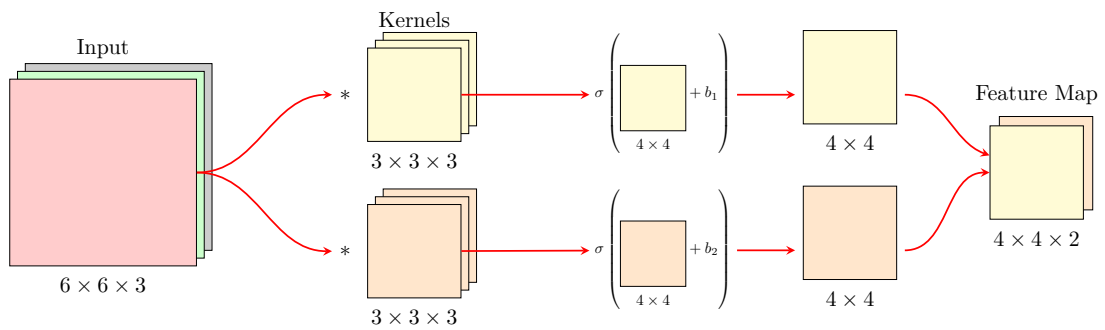
A pooling layer in a neural network summarizes a set of adjacent units from feature map of the preceding layer with a single value. It also uses a kernel to select input regions to summarize it, however the operation does not apply trainable weights. We can see the pooling as a subsampling operation, it has the additional advantage of reducing the volume of data being processed (GONZALEZ; WOODS, 2018). Moreover, we can view the pooling as adding an infinitely strong prior that the function the layer learns must be invariant to small translations. There are two common ways of pooling: average-pooling and max-pooling. The sequence of convolutional and pooling layers is frequently referenced as convolutional block. Its output is the final feature map, which contains input features that are relevant to solve the problem. Frequently, the final feature map is flattened to serve as input to a dense layer.

Figure 7 – Example of a CNN.



Source: Author.

Figure 8 – A convolution layer with two kernels, no padding and a stride value of 1.



Source: Author.

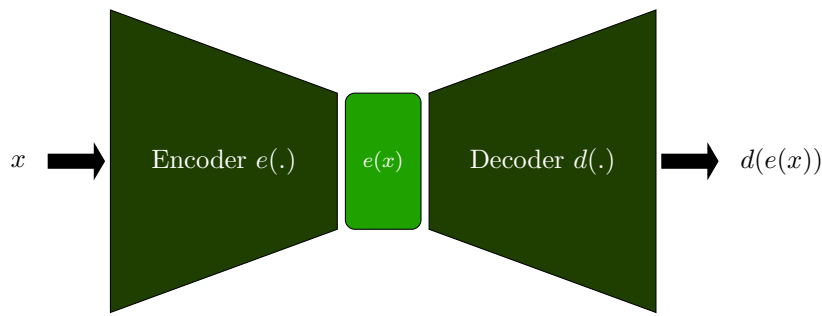
Convolution leverages three important ideas that can help improve a machine learning system: sparse interactions, parameter sharing and equivariant representations (GOODFELLOW; BENGIO; COURVILLE, 2016). CNNs frequently apply kernels which are significantly smaller than the input being processed. This is enough to extract meaningful features, such as edges and textures. Such a behavior results in less weight storage and it is known as sparse interactions. Furthermore, we know that a kernel slides over the entire input. This property is called parameter sharing and further reduces the storage requirements of the network. Finally, the equivariant representations defines a behavior caused by the parameter sharing. As we use a single kernel to extract a specific feature over different regions of the input, we say that CNNs are equivariant to translation.

3.1.2 Autoencoders

An autoencoder is an ANN that is trained to attempt to copy its input to its output, learning an approximation of the identity function (GOODFELLOW; BENGIO; COURVILLE, 2016). Therefore, it is a unsupervised deep learning algorithm, given that

we do not need labeled data. Presently, autoencoders find application in diverse domains, including data denoising and dimensionality reduction. Figure 9 illustrates the usual architecture of an autoencoder. It contains two parts: an encoder $e(\cdot)$ and a decoder $d(\cdot)$. The encoder maps a input x for a representation $e(x)$ belonging to another dimensional space. Whereas the decoder maps such a representation to the output $d(e(x))$, which must be similar to the original input x . The intermediate representation $e(x)$, also called as latent space or encoding, usually have relevant features that describes the input x . Such a property is due to many improvement methods for autoencoders that perform types of processing in the encoder component, making the encoded data different from the input data as much as possible (LI; PEI; LI, 2023). If an autoencoder succeeds in simply learning to set $d(e(x)) = x$ everywhere, then it is not especially useful (GOODFELLOW; BENGIO; COURVILLE, 2016). A common approach is restrict the learning in order to unable the autoencoder to learn to copy perfectly.

Figure 9 – Autoencoder architecture.

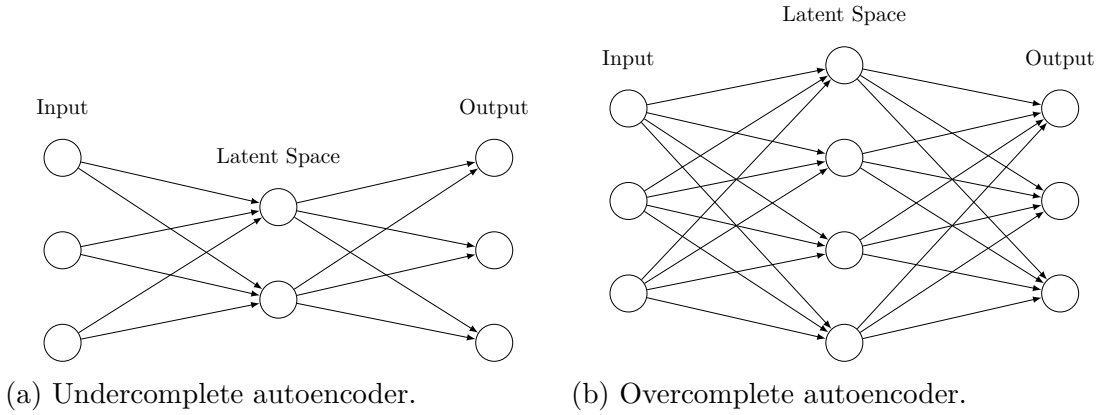


Source: Author.

In terms of the latent space dimensionality, we can classify autoencoders into undercomplete and overcomplete categories. In Figure 10, we illustrate both using a simple architecture for the sake of clarity. Undercomplete autoencoders compute a latent space of smaller dimension than the input data. Learning an undercomplete representation forces the autoencoder to capture the most relevant features from the training data. On the other hand, overcomplete autoencoders compute a latent space of greater dimension than the input data. This can contribute to computing more detailed and complex encodings for feature extraction tasks. However, overcomplete autoencoders have too much capacity, which can lead to learning the copying task learning without extracting useful information about the data distribution.

In case that our problem deals with high complexity data, it is frequently desirable that we apply autoencoders with high latent space dimension. One way to succeed overcomplete autoencoders is applying regularization methods through the loss function. Regularized autoencoders employ a loss function that allows the model to learn additional properties beyond merely replicating its input to its output. Some properties examples are sparsity of the representation, robustness to noise, and to missing inputs.

Figure 10 – Autoencoder classification in terms of the latent space dimensionality.



Source: Author.

In the following, we discuss some improved versions of autoencoders. The denoising autoencoder (VINCENT et al., 2008) is an extension of the traditional autoencoder. It adds a noise signal to the original input, resulting in a corrupted input. It attempts to reconstruct the original noise-free input. Such an autoencoder can ignore the noise signal from the input and extract a more robust intermediate representation. The sparse autoencoder (NG et al., 2011) applies a sparsity constraint to make the activation of the neurons satisfy a certain degree of sparsity. In the original paper, the network achieves such a property by adding a Kullback–Leibler divergence term to the loss function.

An important application of autoencoders is data generation, which we can perform through variational autoencoders (VAEs) (KINGMA; WELING, 2019). VAEs represent a specialized type of autoencoder designed with a regularization approach to mitigate overfitting and guarantee desirable properties in the latent space for generative purposes. Unlike traditional autoencoders, VAEs encode inputs as distributions rather than single points, typically following a normal distribution, thereby enforcing regularization in the latent space. This property is extensively useful in data generation tasks, wherein a random sample from the latent probability distribution is obtained and fed into the decoder to generate an output within the original input data space.

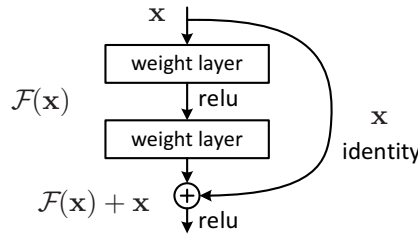
3.2 ANN architectures

In this section, we present some examples of ANN architectures besides the traditional MLP. The networks highlighted below have important mechanisms and concepts that we use in the proposals of this work.

- a) **ResNet**. Due to the degradation problem, ANNs have difficulties in approximating identity mappings by multiple nonlinear layers. He et al. (2016) propose a network, called ResNet, with a special mechanism to address the degradation

problem linked to deep networks: residual blocks. A residual block, see Figure 12, has a residual connection (also called skip connection or shortcut connection). It performs an identity mapping, and its output is added to the outputs of stacked layers. If an identity mapping is an optimal solution for these stacked layers, its weights may tend to zero at the end of training step;

Figure 12 – Residual Block.



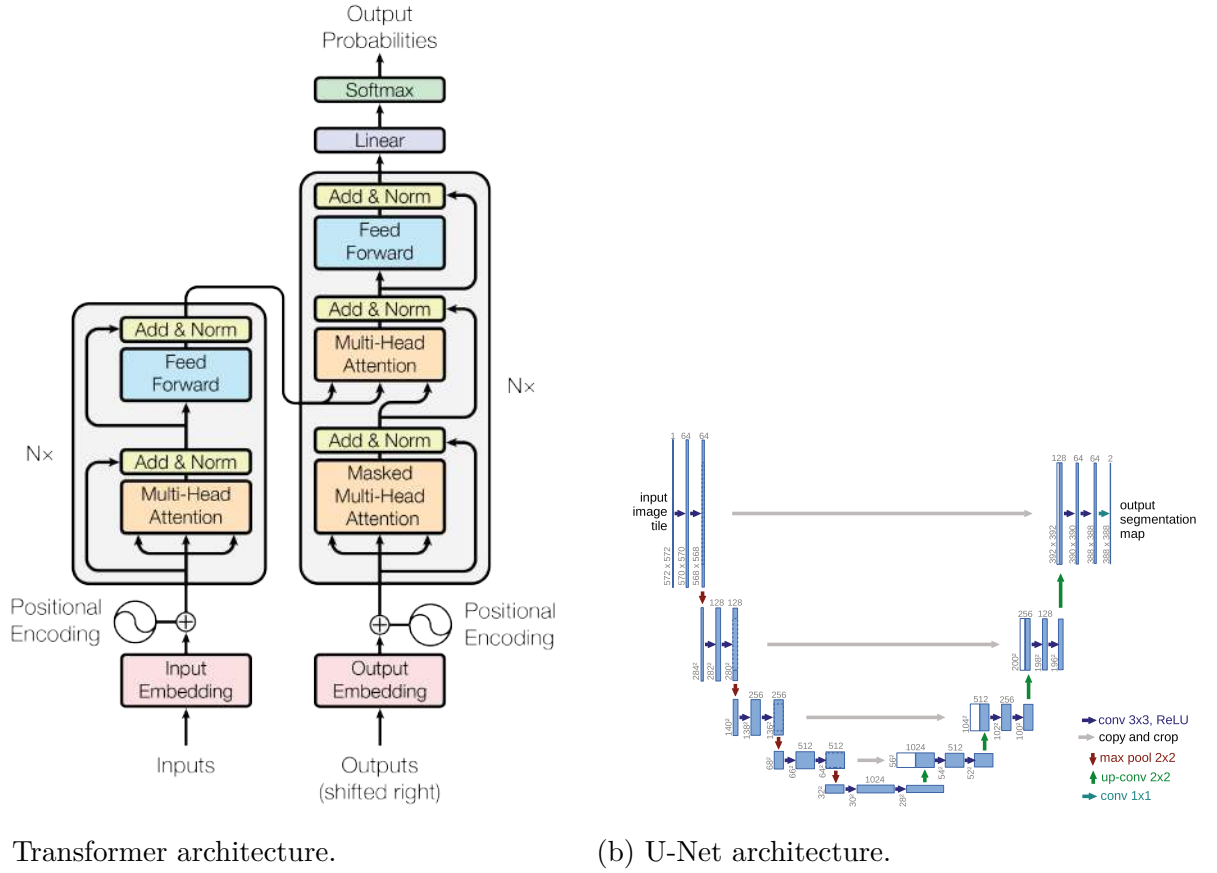
Source: He et al. (2016)

- b) **Transformer.** Vaswani et al. (2017) propose the Transformer architecture, see Figure 13, to address the English-to-German translation task. It relies on the Scaled Dot-Product Attention mechanism to draw global dependencies between input and output. Multi-head attention is a block that performs such mechanism a certain number of times, called heads, in parallel. This enables the network to simultaneously focus on information from various representation subspaces at different positions;
- c) **U-Net.** To address the biomedical image segmentation task, Ronneberger, Fischer e Brox (2015) propose the U-net architecture, see Figure 13. It is a convolutional neural network known for its "U" shape, consisting of a contracting path (encoder) to capture context and an expanding path (decoder) to enable precise localization. This architecture excels at applying contextual information from different resolutions by combining features from deep and shallow layers, resulting in detailed and accurate segmentations;
- d) **Sequential.** In this work, we refer to a network as a sequential architecture, one that does not have any of the special mechanisms presented, i.e., residual connection and multi-head attention. This definition is important for understanding the evaluations that we perform in this work.

3.3 Training and evaluation steps

Data plays a crucial role in ensuring the efficacy of a ML model. Through the data seen in the training step, the ML model can make predictions on real-world tasks. However, to make a fair assessment of its results, we must analyze its generalizability using unseen

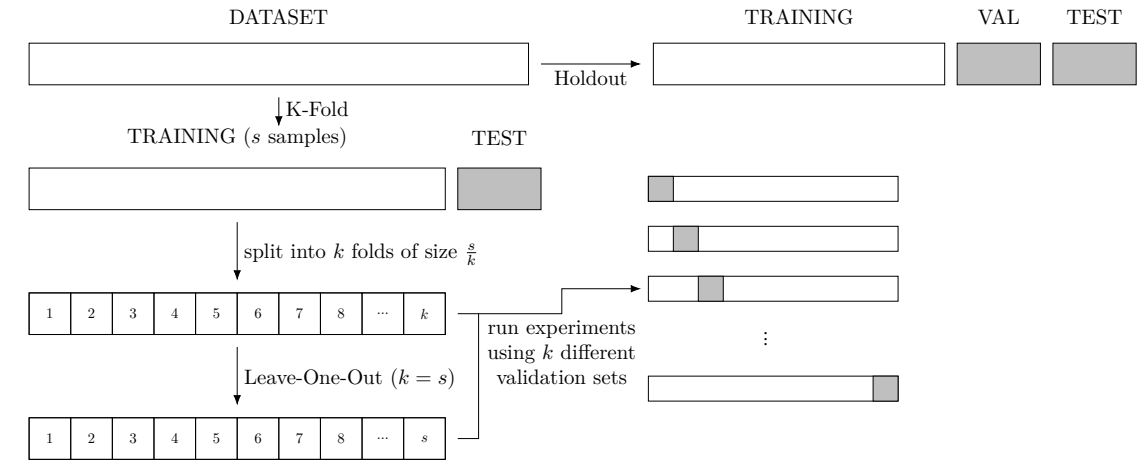
Figure 13 – Examples of famous ANNs architectures.



Source: Vaswani et al. (2017). Ronneberger, Fischer e Brox (2015)

data. Such a process is known as cross-validation and has several types, but we highlight three in this work: Holdout, K-Fold, and Leave-One-Out (LOO). We present each one in Figure 15.

Figure 15 – Cross-validation methodologies.



Source: Author.

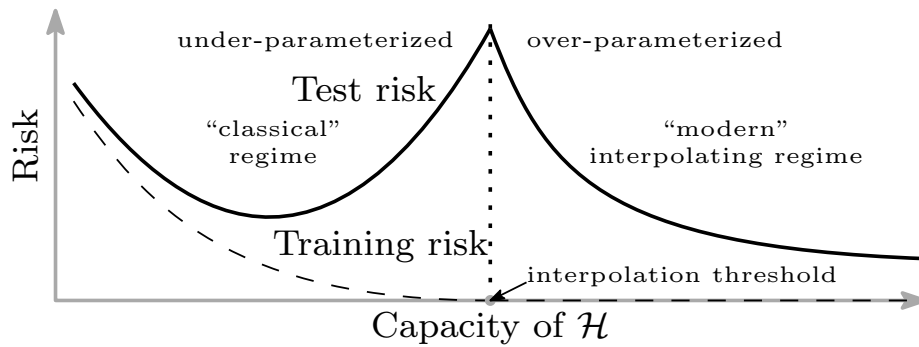
In Holdout cross-validation, we split the data in three sets: training, validation,

and test. The training set is instrumental in share knowledge to the ML model, enabling it to make predictions. However, evaluating the model's performance using the training set is inherently biased, given that the model has already encountered these samples. Consequently, the validation data assumes a pivotal role in evaluating the model during the training phase, facilitating adjustments to hyperparameters and ensuring robustness. Finally, the test data serves as the criterion in the unbiased evaluation process post-training, employing samples that did not influence the model's learning process. In some scenarios, one might don't have enough data to make all three of these data sets. We can squeeze more out of the data using a technique called K-fold cross-validation. In this methodology, we split the data in k sets of equal sizes. Therefore, we perform k rounds of training, where each one applies a different set as the validation set and the remaining as the training set. If the chosen k value is equal to the number of samples, such process is called Leave-One-Out.

Another important step is to select which hypothesis, or model, $\mathcal{H}$ that will be applied in the ML problem. The Figure 16 shows the double-descent risk curve (BELKIN et al., 2019), which we can analyse in two parts: "classical" regime and "modern" interpolating regime. Regarding the capacity (i.e., number of parameters) of hypothesis $\mathcal{H}$, it is more likely to learn the patterns present in the training data. However, the U-shaped test curve from the "classical" regime points out that, after a certain capacity, the learning from the hypothesis cannot be extrapolated to the test. One way to analyze such a behavior is by the bias they impose (regardless of the training dataset) and the variance they produce (from one training set to another). Simpler or more restrictive hypothesis have a greater tendency to induce a greater bias, while for variance the behavior is the opposite. We say a hypothesis is underfitting when it can not learn the necessary patterns from the training data. While a hypothesis is overfitting when it "memorized" the training data patterns, being unable to generalize this learning to unseen data. We can see that there is a trade-off between bias and variance which must be taken into consideration when choosing the hypothesis and its hyperparameters. However, analyzing the "modern" interpolating regime, the test risk starts to decrease after a certain capacity called interpolation threshold. This curve is the result of observations that hypotheses trained to interpolate the training data still obtain near-optimal test results.

Relying the problem that one want to solve, an ANN may need to consume a large amount of samples to perform a decent learning. However, one may not have the amount of data or computing resources needed. Transfer learning is a deep learning technique which consists to copy over the weights learned for task A to a network that will be trained for task B, instead of randomly initializing the weights. This process can take advantage of knowledge acquired for task A that is also relevant for task B, and so may require a reduced amount of data to perform the training step. Such technique is a very popular choice in deep learning community mainly due to the large number of models trained on

Figure 16 – The double-descent risk curve.



Source: Belkin et al. (2019)

considerable amounts of data that are publicly available.

3.4 Machine Learning and Deep Learning

Deep learning (DL) is a broad family of techniques for machine learning in which hypotheses take the form of complex network architectures with tunable connection strengths. The word “deep” refers to the fact that the networks are typically organized into many layers, which means that computation paths from inputs to outputs have many steps. This allows complex interactions between all input variables.

More recently, many ML projects have been applied DL approaches in order to compute the solution. According to Goodfellow, Bengio e Courville (2016), this has happened due to 3 main factors: the large amount of stored data, the construction of more powerful computers and the development of techniques to efficiently train deeper neural networks. This depth in DL is responsible for a efficient representation learning by computing representations that are expressed in terms of other, simpler representations. However, this depth requires a high amount of data and high computing power, requirements that are not normally necessary for other traditional machine learning models.

Deep learning can perform the representation learning by consuming directly the raw input data. Most machine learning approaches receives features that were computed by feature engineering using human-engineered techniques. Such techniques are sometimes costly or not even possible to develop. This is a great advantage of DL, given that they perform this feature engineering process automatically through a refinement of representation learning throughout the layers. Such a property gives the possibility for a DL model to be end-to-end (E2E) learning: the only component in a learning system that represents the entire pipeline. Furthermore, some other machine learning models have an advantage in explainability, such as decision trees. Given an output computed by a deep ANN, we can not directly extract an explanation for this computation.

4 Proposals

In this chapter, we present our contributions. Firstly, we propose five baseline models for evaluating retargeted image quality. Secondly, we present the AEIMPS, our autoencoder-based IRQA that computes the quality in a metric fusion fashion. Then, we detail the EMPIRQ framework, which was inspired by AEIMPS, elucidating its structure and the various combinations explored.

4.1 Baselines

Comparing IRQAs is an effective way to determine which performs better. Selection of the most suitable IRQA can be based on statistical performance scores. However, this approach does not guarantee that the chosen IRQA is truly optimal, as its underlying behavior is often unknown or difficult to model. Thus, comparing these IRQAs against a set of predictable behavior models, referred to as baselines, is beneficial for assessing whether the proposed metrics are adequate for a specific application. To construct these baselines, we utilized the same CUHK dataset and evaluation methodology employed in the IRQAs discussed in Section 2.3.

The proposed baseline models can be classified into two main methodologies: human-based and statistical-based. The human-based baselines utilize the evaluations provided by individuals who contributed to the dataset construction. Specifically, the individual quality scores assigned by each participant are used to compute this baseline. Once these scores have been regressed using the logistic function from Equation 2.1, we calculate the four statistical performance scores for each individual. This approach provides a way to assess the behavior of the model by directly leveraging human input.

The statistical-based baselines rely on statistical distributions, random sampling methods, and a heuristic approach. By utilizing these baselines, we are able to assess whether the evaluated IRQA performs worse than a biased model, which would suggest that it may not be suitable for the intended application. This comparison provides valuable insights into the relative effectiveness of the IRQA. The properties and characteristics of the five proposed baseline models are discussed in greater detail below:

- a) **Expert.** This baseline utilizes the scores provided by the human subject who demonstrated the highest correlation with the Mean Opinion Score (MOS) values. The primary objective of this baseline is to establish a benchmark for IRQAs that exhibit a high level of agreement with the MOS, effectively simulating the behavior of an "expert" human evaluator. By doing so, it sets

a reference point for assessing the performance of IRQA methods that closely align with expert-level human judgment;

- b) **Amateur**. Similar to the **Expert baseline**, this approach uses the scores from the human subject who exhibited the lowest correlation with the MOS values. The primary aim is to simulate the behavior of a “lay” person with minimal expertise in IRQA. This baseline serves as a reference point for evaluating how an IRQA might perform when compared to the judgment of a non-expert with limited understanding of image quality assessment;
- c) ρ . This baseline relies solely on the retargeting ratio value, as defined in Equation (4.1), which is applied to resize an image $\mathbf{I}$ to $\mathbf{J}$, yielding a median efficiency. The underlying assumption is that the greater the reduction in image size, the higher the probability of distorting its content or structure, or introducing visual artifacts, thereby degrading the overall quality of the image. This baseline captures the relationship between image reduction and potential quality loss, serving as a reference for evaluating the impact of resizing on IRQA performance;
- d) **Permutation**. This baseline employs a random-based approach, which involves random sampling with replacement of the MOS values over a specified number of iterations. By using this method, we aim to explore how random variations can capture viewing or assessment biases that are commonly observed across different images. This baseline helps in understanding the influence of random sampling on the evaluation process and provides a reference for assessing the robustness of the IRQA;
- e) **Chance**. This baseline involves sampling random values from the MOS distribution, which has been fitted using a Gaussian Mixture Model. The objective of this approach is similar to that of the **Permutation** baseline, aiming to capture variations in the MOS distribution. To achieve this, a specified number of iterations must be performed. This baseline helps in understanding how random sampling from a fitted distribution can affect the evaluation process and provides insights into the variability and robustness of the IRQA.

$$\rho(\mathbf{I}_{m \times n}, \mathbf{J}_{m' \times n'}) = \frac{\min(m, m') \min(n, n')}{\max(m, m') \max(n, n')} \quad (4.1)$$

4.2 AEIMPS

In the work of Carvalho e Oliveira (2022), we introduced our initial proposal for an autoencoder-based IRQA: textbfAutoEncoder Information MaP Similarity (AEIMPS). This method integrates various types of information using an autoencoder in a metric fusion fashion, as illustrated in Figure 17. The data utilized includes the original and

retargeted images, $\mathbf{I}$ and $\mathbf{J}$, their respective encodings, $e(\mathbf{I})$ and $e(\mathbf{J})$, the reconstructed retargeted image, $d(e(\mathbf{J}))$, and the retargeting factor, $\rho(\mathbf{I}, \mathbf{J})$. The development of AEIMPS was primarily driven by two key motivations:

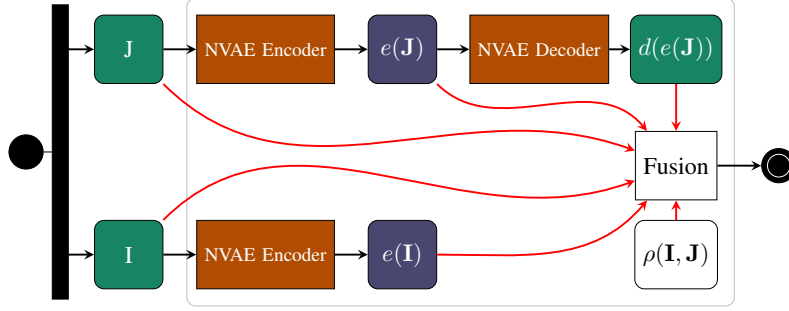
- a) We assumed that the latent space of the autoencoder encapsulates significant features that represent the content of the image. In this framework, a satisfactorily retargeted image would have a latent code that closely resembles that of the original image. Conversely, an unsatisfactorily retargeted image would have a latent code that “lacks” or fails to preserve these valuable features. This assumption underpins the rationale for using latent space representations to evaluate the quality of image retargeting;
- b) We anticipated that the reconstructed image would exhibit some distortion but would be ‘restored’ to approximate the original, i.e., $d(e(\mathbf{J})) \approx \mathbf{I}$. Thus, AEIMPS aims to leverage the reconstruction capability of autoencoders, utilizing this effect to assess image quality. By comparing the reconstructed image to the original, AEIMPS can potentially benefit from the autoencoder’s ability to highlight distortions and quality degradation in retargeted images.

The autoencoder architecture selected for AEIMPS is NVAE (VAHDAT; KAUTZ, 2020), which was originally designed for data generation tasks. NVAE is a deep hierarchical variational autoencoder that utilizes depth-wise separable convolutions and batch normalization. Specifically, it employs separable convolutions in the depth direction for model generation, while using standard convolutions for the encoder model. NVAE incorporates a residual parameterization of normal distributions, and its training is enhanced through spectral regularization. For our implementation, we utilize the official NVAE code repository available on GitHub¹. NVAE is available in various versions trained on different datasets; we opted for the version trained on the CelebA 64 dataset.

For the metric fusion, we integrate several information sources—namely $e(\mathbf{J})$, $e(\mathbf{I})$, $d(e(\mathbf{J}))$, and $\rho(\mathbf{I}, \mathbf{J})$ —into a single score. This fusion is achieved using difference scores, such as the complement of mean absolute error (CMAE). We explored 13 different fusion strategies (see Appendix A), including methods such as simple averaging, product, linear addition, and non-linear combinations. The results obtained with AEIMPS are intermediate (see Table 3); our best outcome was achieved through a linear combination of the CMAE score between the image and its reconstruction from NVAE and the retargeting factor, specifically: $0.2 \cdot \text{cmae}(\mathbf{I}, d(e(\mathbf{J}))) + 0.8 \cdot \rho(\mathbf{I}, \mathbf{J})$. We believe that using other autoencoder architectures trained for tasks beyond data generation, such as NVAE, could potentially enhance the results. Despite the intermediate outcomes, we continue to pursue this research, as we consider that an autoencoder-based IRQA holds considerable promise.

¹ NVAE code repository: <https://github.com/NVlabs/NVAE>

Figure 17 – AEIMPS flowchart.



Source: Carvalho e Oliveira (2022)

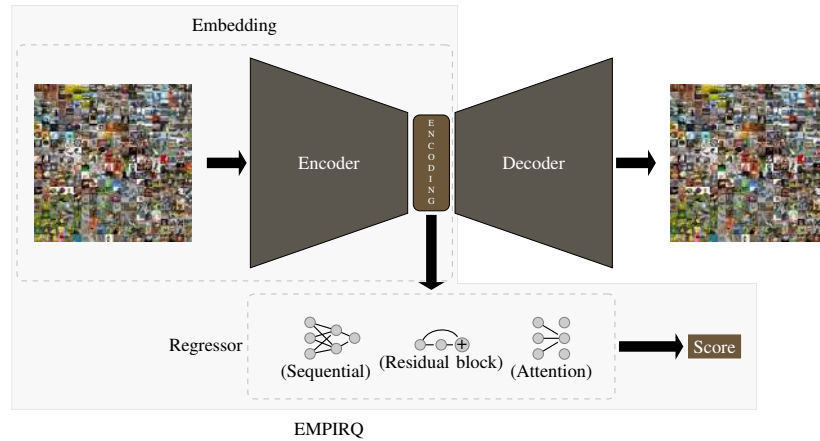
4.3 EMPIRQ

Despite utilizing a relatively simple approach for assessing image retargeting quality, AEIMPS achieved only intermediate results. Given that autoencoders are an effective method for learning and extracting significant features (LI; PEI; LI, 2023), we are committed to further developing an autoencoder-based IRQA to better exploit the latent space. To enhance the quality score computation, we propose a no-reference approach that incorporates a more robust strategy rather than relying solely on metric fusion. Additionally, this new approach does not utilize the original image, the reconstructed image, or the resizing factor.

We introduce our no-reference IRQA proposal, **Encoding MaP Image Retargeting Quality** (EMPIRQ), as illustrated in Figure 18. EMPIRQ consists of two main components designed to compute a quality score for the image retargeting problem. The first component is the embedding computer, which extracts relevant features from the input image. The second component is a regressor that takes this embedding as input and calculates the quality score for the image. This approach aims to evaluate image retargeting quality without relying on reference images or any additional information, as retargeting factors.

The encoder component of the autoencoder architecture is designed to extract meaningful features necessary for image reconstruction by the decoder component. We leverage this property in our pipeline by utilizing the encoder components of publicly available autoencoders. Importantly, we do not employ the decoder or retrain these networks; instead, we reuse their pre-trained weights, which were derived from extensive datasets. The primary objective is to investigate whether this previously acquired knowledge can be applied to the image retargeting quality assessment problem. For EMPIRQ, we continue to use the NVAE architecture. However, we also aim to explore additional autoencoder networks trained for other tasks. To this end, we have incorporated the Evolutionary Autoencoders from Suganuma, Ozay e Okatani (2018) into our approach.

Figure 18 – EMPIRQ flowchart.



Source: Author.

Suganuma, Ozay e Okatani (2018) propose a straightforward convolutional autoencoder that utilizes standard network components, including convolutional layers and skip connections. However, their approach includes an evolutionary algorithm to search for the most suitable architectures. We have four main motivations for selecting the Evolutionary Autoencoder as the second candidate for EMPIRQ. First, it is trained on the CelebA 64 dataset, similar to NVAE, which allows for a more equitable comparison. Second, its pre-trained weights are publicly available. Third, we did not encounter memory issues during its use in our experiments. Finally, unlike NVAE, which is designed for data generation, the Evolutionary Autoencoder is trained for the denoising task. This enables us to explore the impact of using autoencoders trained for different tasks in our pipeline. We utilize its official code repository available on GitHub².

The regressor component in EMPIRQ maps the retargeted image embedding to its corresponding quality score. To achieve this, we employ some ANNs architectures. Notably, this is the only component of our pipeline where we perform the training step. For the regressor candidates, we have selected one from the following ANN architectures: sequential, with residual blocks, or incorporating attention mechanisms. By leveraging such different architectures, we aim to assess their impact on EMPIRQ's performance by identifying the most effective for the IRQA task. Consequently, we evaluate six combinations between embeddings and architectures. Next, we provide detailed descriptions of each architecture:

- a) **Sequential.** CONV2D $\rightarrow$ FLATTEN $\rightarrow$ DENSE $\rightarrow$ DENSE;
- b) **Residual.** RESNETBLOCK $\rightarrow$ FLATTEN $\rightarrow$ DENSE $\rightarrow$ DENSE;
- c) **Attention.** MULTIHEADATTENTION $\rightarrow$ CONV2D $\rightarrow$ FLATTEN $\rightarrow$ DENSE $\rightarrow$ DENSE;

² Evolutionary Autoencoders code repository: <https://github.com/sg-nm/Evolutionary-Autoencoders>

5 Experiments and results

This chapter outlines the experimental framework used for evaluating our proposals, presents the results obtained, and includes a discussion of these results. It details the methodologies employed, the outcomes of our experiments, and provides an analysis of the findings to offer insights into the performance and effectiveness of the proposed approaches.

5.1 Experiments

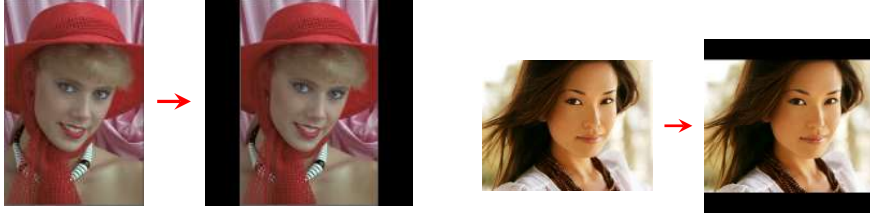
The experiments were conducted using the Python programming language. TensorFlow and Keras were employed to build and manage the regressor architectures, while the autoencoder architectures were implemented using the original code provided by the authors, which is based on PyTorch. The development environment utilized was Google Colab, with computations performed on an NVIDIA T4 GPU. To compute the EMPIRQ, AEIMPS and baseline results, we applied the IRQA evaluation methodology outlined in section 2.3. Since our focus is on predicting the MOS values, we selected the CUHK dataset, which provides MOS annotations, for our analysis.

In the following, we provide a more detailed explanation of the CUHK dataset methodology. The CUHK dataset consists of 57 original images, each depicting diverse contexts, such as human faces or landscapes. Each original image is accompanied by three retargeted versions, where each version is resized to either 50% or 75% of its original width using one of the ten retargeting operators described in section 2.1. As a result, the dataset includes a total of 171 retargeted images. Evaluators were divided into two sessions, each consisting of different sets of images. In the first session, 30 participants rated the retargeted images observing the reference and retargeted image, while in the second session, 34 participants performed the evaluations only observing the retargeted image. Among the participants, some had expertise in image processing, with 15 experts in the first session and 18 experts in the second session.

We evaluated six combinations of encoding embeddings with regressor architectures in EMPIRQ. To ensure compatibility with the input shape requirements of the NVAE and Evolutionary Autoencoders architectures, we resized the images using padding to fit the corresponding input dimensions, employing the bilinear interpolation method, as shown in Figure 19. Additionally, the images were normalized to a range between 0 and 1. It is important to note that only the regressor component of EMPIRQ underwent training. All regressors were trained using the Adam optimizer for 100 epochs, with a batch size of 4 and a learning rate of 0.001. The loss function used was mean squared error. Early Stopping was implemented with a patience of 30 epochs, and the best model was selected

based on the validation loss.

Figure 19 – Resizing with padding.



(a) Vertical padding resizing.

(b) Horizontal padding resizing.

Source: Carvalho e Oliveira (2022)

We trained the regressor architectures in EMPIRQ using the corresponding data division, which consists of image embeddings computed by the encoders from either NVAE or Evolutionary Autoencoders. After training, the predicted MOS values were calculated for all samples in the dataset and transformed using the five-parameter logistic function described in Equation (2.1). Subsequently, the four statistical scores—LLC, SRCC, RMSE, and OR—were computed using these transformed predicted MOS values and the MOS values provided in the dataset. This experimental pipeline undergoes minor adjustments depending on the cross-validation methodology employed, as detailed in the following points.

- a) **K-Fold**. The dataset is divided into 10 folds. We perform the training process 10 times, with each fold serving as the validation set once while the remaining nine folds are used for training. The model that achieves the best statistical scores across these iterations is selected;
- b) **Hold-out**. We use holdout validation with two different splits: 80/20 and 50/50 for training and validation data, respectively. For each split, the training process is repeated 30 times, and statistical scores are computed using the entire dataset. Models that fail to converge during training, potentially due to issues such as random weight initialization or local minima, are excluded from consideration. The final result for this approach is the mean of the statistical scores from the remaining models;
- c) **Leave-One-Out**. In this approach, each training iteration uses one sample as the validation data while the remaining samples serve as the training data, resulting in 171 models. If a model fails to converge during training, the process is repeated using the same data division. The predicted MOS for each sample is obtained from the model in which that sample was used as the validation data. Consequently, statistical scores are computed based on these predicted MOS values;

- d) **Leave-One-Group-Out.** This methodology is similar to the Leave-One-Out approach, with the key difference being that the validation data in each iteration consists of a group of images derived from the same original image. The primary goal of this approach is to mitigate the influence of training samples that are similar to those in the validation set. By doing so, we aim to ensure that the model can generalize more effectively to unseen data.

With these results, we conducted two types of analyses: one quantitative and one visual. The quantitative analysis involves comparing all combinations of encoding embeddings with regressor architectures, focusing on their statistical measurements. The visual analysis involves comparing these combinations against the baselines described in Section 4.1. Additionally, we include AEIMPS and the IRQAs listed in Table 1 in this analysis. The findings from these analyses are explored in the following sections.

5.2 Results and discussion

In this section, we present and discuss the findings from the experiments. We begin by analyzing the performance of different EMPIRQ architecture combinations. Next, we introduce the metrics computed from the proposed baselines and compare these with the results from the EMPIRQ combinations. Finally, we compare the performance of EMPIRQ, AEIMPS, the baselines, and state-of-the-art IRQAs.

5.2.1 Encoding embeddings with regressor combinations analysis for EMPIRQ

The results for the six combinations of encoding embeddings with regressor architectures in EMPIRQ, evaluated using each cross-validation methodology, are presented in Table 2. Overall, the Evolutionary Autoencoder consistently achieved better performance metrics compared to NVAE. Despite the fact that the Evolutionary Autoencoder is simpler in design—being a standard convolutional autoencoder—whereas NVAE is a more complex variational autoencoder utilizing depth-wise separable convolutions and batch normalization—the Evolutionary Autoencoder’s superior performance suggests that autoencoders trained for denoising tasks may be more effective in the EMPIRQ pipeline compared to those trained for data generation tasks.

Among the regressor options, the ones with residual blocks obtained the most promising results. Since residual connections enable the reuse of learned features from earlier layers, thus, making it easier for the network to learn new features in later layers, we believe this encouraged the network to create more abstract and complex representations by combining both low-level and high-level features. Also, we support that by capturing such abstract and complex representations, this architecture somehow learn the complexity of distortions, thus, assessing more accurately the results. Another interesting finding is

Table 2 – Results of each encoding embeddings with regressor combination for EM-PIRQ.

Encoder	Regressor	Methodology	LLC	SRCC	RMSE	OR
NVAE	 (Sequential)	Holdout (80/20)	0.82 ± 0.05	0.84 ± 0.05	8.53 ± 1.42	0.03 ± 0.02
		Holdout (50/50)	0.70 ± 0.06	0.71 ± 0.06	9.99 ± 0.93	0.02 ± 0.01
		K-Fold	0.89 ± 0.04	0.90 ± 0.04	6.97 ± 1.42	0.03 ± 0.02
		Leave-One-Out	0.96	0.97	4.21	0
		Leave-One-Group-Out	0.83	0.83	7.81	0
	 (Residual block)	Holdout (80/20)	0.88 ± 0.05	0.89 ± 0.05	6.57 ± 1.32	0.02 ± 0.01
		Holdout (50/50)	0.78 ± 0.02	0.77 ± 0.02	8.61 ± 0.45	0.02 ± 0.01
		K-Fold	0.93 ± 0.04	0.93 ± 0.04	6.13 ± 1.64	0.04 ± 0.03
		Leave-One-Out	0.94	0.95	4.95	0.03
		Leave-One-Group-Out	0.83	0.84	7.82	0
	 (Attention)	Holdout (80/20)	0.74 ± 0.09	0.74 ± 0.09	9.37 ± 1.41	0.01 ± 0.01
		Holdout (50/50)	0.65 ± 0.08	0.65 ± 0.09	10.45 ± 0.85	0 ± 0.01
		K-Fold	0.74 ± 0.09	0.74 ± 0.10	9.53 ± 1.55	0.01 ± 0.01
		Leave-One-Out	0.93	0.95	5.40	0.01
		Leave-One-Group-Out	0.81	0.81	8.18	0
Evolutionary	 (Sequential)	Holdout (80/20)	0.84 ± 0.07	0.84 ± 0.07	7.60 ± 1.45	0.01 ± 0.01
		Holdout (50/50)	0.72 ± 0.05	0.72 ± 0.05	9.41 ± 0.70	0.01 ± 0.01
		K-Fold	0.84 ± 0.14	0.83 ± 0.14	7.35 ± 2.54	0.01 ± 0.01
		Leave-One-Out	0.90	0.92	6.28	0.01
		Leave-One-Group-Out	0.76	0.78	9.00	0.01
	 (Residual block)	Holdout (80/20)	0.92 ± 0.02	0.92 ± 0.02	5.69 ± 0.86	0.01 ± 0.01
		Holdout (50/50)	0.80 ± 0.02	0.79 ± 0.02	8.48 ± 0.53	0.01 ± 0.01
		K-Fold	0.94 ± 0.05	0.94 ± 0.05	5.55 ± 1.84	0.02 ± 0.02
		Leave-One-Out	0.98	0.98	3.06	0.01
		Leave-One-Group-Out	0.82	0.83	8.29	0
	 (Attention)	Holdout (80/20)	0.59 ± 0.17	0.59 ± 0.17	12.82 ± 9.04	0.06 ± 0.22
		Holdout (50/50)	0.56 ± 0.07	0.56 ± 0.08	11.31 ± 0.76	0.00 ± 0.00
		K-Fold	0.65 ± 0.10	0.65 ± 0.09	11.25 ± 1.86	0.01 ± 0.01
		Leave-One-Out	0.90	0.92	6.99	0
		Leave-One-Group-Out	0.74	0.75	9.57	0

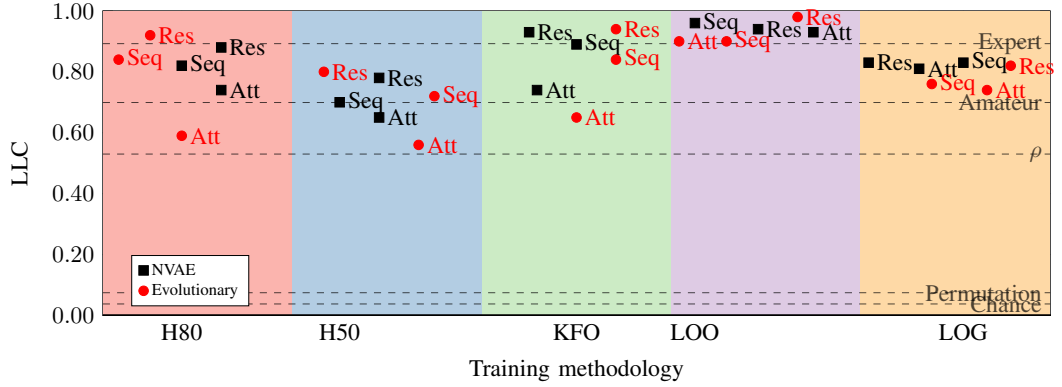
Source: Author.

related to the attention model combinations. We noticed that in most cases of Holdout and K-Fold training methodology it performed poorly against the others. Diverse datasets are generally beneficial for training attention-based models, but our dataset is very limited, thus, impairing the ability of attention mechanisms to learn abstract and complex representations. Therefore, our best IRQA combination is the evolutionary encoder and regressor architecture with residual blocks.

5.2.2 Baseline analysis

In Figure 21, we present the results of all combinations from Table 2, specifically focusing on the LLC metric and grouped according to the five cross-validation methodologies discussed earlier. We opted to visualize only the LLC metric due to its strong correlation with other metrics. From this visualization, it is clear that all combinations outperform the ρ baseline and, consequently, also surpass the permutation and chance baselines. This suggests that our proposed model performs more effectively than both biased and random models. Additionally, it is evident that all combinations with performance below the human amateur level include the attention-based regressor in their pipeline. This observation indicates that the attention-based regressor may contribute to lower performance within the EMPIRQ framework, particularly when applied to the CUHK dataset.

Figure 21 – LLC metric for all encoding embeddings $\times$ regressor combinations grouped by cross validation methodology.



Source: Author.

Additionally, 87% (26 out of 30) of the combinations fall within the spectrum of human evaluation, surpassing the human amateur level. Notably, 30% (9 out of 30) of these combinations achieved 'superhuman' scores, exceeding even those of human experts. This suggests that EMPIRQ has the potential to perform at a level comparable to that of an average human evaluator. Moreover, depending on the specific component configuration within its pipeline, EMPIRQ can achieve performance on par with human experts. Our analysis identified that the combination involving the Evolutionary ResNet-based model, trained using the Leave-One-Out methodology, achieved the highest ranking. Consequently, we used this top-performing configuration for comparison against other state-of-the-art models in the subsequent evaluations.

5.2.3 CUHK benchmark analysis

In Table 3, we present the statistical scores of various metrics obtained from (ASHEGHI et al., 2022). These scores are ranked from best to worst based on the LLC. The TYPE column indicates the IRQA category in relation to the type of data consumed by the algorithm. Furthermore, we have also included the scores for the four proposed baseline models as well as our best EMPIRQ and AEIMPS methods, allowing for a consistent evaluation with the elements discussed in section 4.1. Out of the 18 models available, 11% are categorized as NR, 22% as RR, and 67% as FR. We believe the preference for FR models may be attributed to the user's desire to leverage all available information.

Our initial finding is that all evaluated IRQA methods outperformed the random baselines, indicating the absence of bias related to the score distribution. Conversely, when examining the ρ baseline, which incorporates the scale factor utilized in the retargeting operation, we found that approximately 33% of the models performed worse than this baseline, highlighting a pronounced bias towards the retargeting ratio. In addition, the overall results reveal that only the NR models, specifically Yan2017 (YAN et al., 2017)

Table 3 – IRQAs comparison by statistical scores with proposals and baselines.

IRQA	TYPE	LLC	SRCC	RMSE	OR
EMPIRQ	NR	0.98	0.98	3.06	0.01
Yan2017 (YAN et al., 2017)	NR	0.921	0.912	3.201	0.011
Expert	-	0.8925	0.8924	6.2093	0.0
INSEM (LI et al., 2021)	FR	0.798	0.748	7.905	0.023
RN-IRQA (NIU et al., 2021)	FR	0.792	0.761	8.234	0.040
Li2019 (LI; GUO; JIN, 2019)	FR	0.774	0.770	8.542	0.070
MLF (ZHANG et al., 2018)	FR	0.757	0.738	8.525	0.029
HCDL (FU et al., 2018)	RR	0.717	0.684	9.135	0.021
Karimi2017 (KARIMI et al., 2018)	FR	0.716	0.709	9.639	0.059
Amateur	-	0.6992	0.6724	9.3655	0.0
ARS (ZHANG et al., 2016)	FR	0.683	0.669	9.855	0.070
HDPM (LIN et al., 2015)	FR	0.671	0.690	-	-
PGDIL (HSU et al., 2013)	FR	0.540	0.540	11.361	0.152
AEIMPS	FR	0.5356	0.4557	11.4007	0.0
ρ	-	0.5299	0.5298	11.4497	0.0
Liu2011 (LIU et al., 2011)	RR	0.437	0.466	12.141	0.152
EH (MANJUNATH et al., 2001)	RR	0.342	0.328	12.686	0.204
SIFT-flow (LIU et al., 2008)	FR	0.314	0.290	12.817	0.146
BDS (SIMAKOV et al., 2008)	FR	0.289	0.288	12.922	0.216
EMD (PELE; WERMAN, 2009)	RR	0.276	0.290	12.977	0.169
IR-SSIM (FANG et al., 2014)	FR	0.230	0.240	13.140	0.176
Permutation	-	0.0746	0.0712	13.4499	0.0002
Chance	-	0.0377	0.0146	13.4913	0.0

Source: Author and Asheghi et al. (2022).

and EMPIRQ, exceeded the Expert baseline. Furthermore, only six other metrics fell within the *human baseline spectrum*, which lies between the Amateur and Expert baselines. Additionally, AEIMPS produced intermediate results. Its placement just above the ρ baseline in the results table is logical, as its optimal configuration is a linear combination where the retargeting factor is assigned a weight of 0.8.

Note that our two IRQA proposals, AEIMPS and EMPIRQ, exhibit different characteristics. AEIMPS is a simpler IRQA method compared to EMPIRQ. However, it does not require any annotated data, as it does not involve a training step. As a result, AEIMPS is a more cost-effective approach that can be applied in resource-constrained contexts. Additionally, in AEIMPS, we only implemented the NVAE. It is possible that even more promising results could be achieved by employing a different autoencoder architecture, while still preserving its simplicity.

Although it is not entirely accurate to assert that NR models consistently outperform FR and RR models, particularly in image retargeting applications, we argue that NR models possess certain inherent advantages. Given that IRQA is inherently subjective, NR models, especially those based on machine learning, are designed to capture perceived

quality aspects without explicit reference to the original image. This characteristic can be beneficial when the aim is to align with human perceptual judgments. Furthermore, by leveraging features (embeddings) from models trained on a broader range of images, NR models are well-positioned to identify the most relevant partial information in IRQA. This broader training spectrum may enhance their ability to discern which aspects of the image are most pertinent to quality assessment. On the other hand, FR models, which rely on direct comparison between the original and retargeted images, and RR models, which use partial information from the reference image, may tend to focus on low-level pixel-wise differences. This pixel-level focus can result in overlooking high-level perceptual quality aspects, thereby failing to accurately capture the nuances of subjective quality judgments.

6 Conclusion

In this work, we proposed a No-Reference Image Retargeting Quality Assessment (NR IRQA) algorithm, called **E**ncoding **MaP** Image **R**etargeting **Q**uality (EMPIRQ), based on encodings computed by autoencoder architectures. Specifically, we evaluated six combinations of autoencoders (NVAE and Evolutionary) and regressor architectures (Sequential, Residual block, and Attention). The regressor architectures were trained using five different methodologies. Such an algorithm was inspired by our previous autoencoder-based IRQA: **A**uto**E**ncoder **I**nformation **MaP** **S**imilarity (AEIMPS). It combines both the latent space and the reconstructed image from NVAE to compute the quality in a metric fusion fashion. Additionally, we proposed five baseline models, namely, Expert, Amateur, ρ , Permutation, and Chance, to analyze IRQA results and investigate some bias towards a few aspects.

Our findings revealed that the Evolutionary combination with the regressor architecture with residual blocks was the best configuration, achieving the top results, surpassing the Expert baseline. We believe this is due to the residual connections with learned features from earlier layers. Also, this suggests potential advantages in using NR models for evaluating quality in retargeted images. Furthermore, we argue that Full-Reference (FR) and Reduced-Reference (RR) models may focus more on low-level pixel-wise differences than high-level perceptual quality aspects, potentially impacting their overall performance. Moreover, the superiority of the Evolutionary Autoencoders' results compared to NVAE may indicate an influence of the type of task for which the autoencoder was originally trained in our EMPIRQ pipeline. The Evolutionary Autoencoders were trained for denoising tasks, whereas NVAE was designed for data generation.

As part of our future work, we are investigating the application of our model in computing Difference Mean Opinion Score (DMOS). This will be particularly valuable for retargeted image datasets where the output is a rank instead of a score, such as RetargetMe and NRID. Moreover, we plan to apply our model to additional datasets to verify its generalization capability and efficiency. Finally, we also plan to explore the possibility of incorporating autoencoders originally trained for others different tasks, into the pipeline. By doing so, we aim to analyze the influences of this aspect in EMPIRQ.

Bibliography

- ASHEGHI, B. et al. A comprehensive review on content-aware image retargeting: From classical to state-of-the-art methods. *Signal Processing*, v. 195, p. 108496, 2022. ISSN 0165-1684. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0165168422000433>>. Citado 3 vezes nas páginas 41, 65, and 66.
- AVIDAN, S.; SHAMIR, A. Seam carving for content-aware image resizing. *ACM TRANSACTIONS ON GRAPHICS*, v. 26, n. 3, JUL 2007. ISSN 0730-0301. ACM SIGGRAPH 2007 Conference, San Diego, CA, AUG 05-09, 2007-2008. Citado na página 33.
- BELKIN, M. et al. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, v. 116, n. 32, p. 15849–15854, 2019. Disponível em: <<https://www.pnas.org/doi/abs/10.1073/pnas.1903070116>>. Citado 2 vezes nas páginas 52 and 53.
- CARVALHO, L.; OLIVEIRA, S. Aeimps: Deep autoencoder for image retargeting quality assessment. In: *Anais Estendidos da XXXV Conference on Graphics, Patterns and Images*. Porto Alegre, RS, Brasil: SBC, 2022. p. 72–75. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/sibgrapi_estendido/article/view/23263>. Citado 4 vezes nas páginas 35, 56, 58, and 62.
- DONG, W. et al. Optimized image resizing using seam carving and scaling. *ACM Trans. Graph.*, Association for Computing Machinery, New York, NY, USA, v. 28, n. 5, p. 1–10, dec 2009. ISSN 0730-0301. Disponível em: <<https://doi.org/10.1145/1618452.1618471>>. Citado na página 34.
- FANG, Y. et al. Objective Quality Assessment for Image Retargeting Based on Structural Similarity. *IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS*, 4, n. 1, p. 95–105, MAR 2014. ISSN 2156-3357. Citado 3 vezes nas páginas 40, 41, and 66.
- FU, Z. et al. Quality assessment of retargeted images using hand-crafted and deep-learned features. *IEEE ACCESS*, v. 6, p. 12008–12018, 2018. ISSN 2169-3536. Citado 4 vezes nas páginas 30, 38, 41, and 66.
- GONZALEZ, R.; WOODS, R. *Digital Image Processing*. Pearson, 2018. ISBN 9780133356724. Disponível em: <<https://books.google.com.br/books?id=0F05vgAACAAJ>>. Citado na página 46.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016. Disponível em: <<http://www.deeplearningbook.org>>. Citado 5 vezes nas páginas 43, 45, 47, 48, and 53.
- HE, K. et al. Deep Residual Learning for Image Recognition. In: *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016. (CVPR '16), p. 770–778. ISSN 1063-6919. Disponível em: <<http://ieeexplore.ieee.org/document/7780459>>. Citado 2 vezes nas páginas 49 and 50.

HSU, C.-C. et al. Objective quality assessment for image retargeting based on perceptual distortion and information loss. In: IEEE; SARAWAK CONVENT BUR; MALAYSIA CONVENT & EXHIBIT BUR; IEEE CIRCUITS & SYST SOC; NEURAMATIX. *2013 IEEE INTERNATIONAL CONFERENCE ON VISUAL COMMUNICATIONS AND IMAGE PROCESSING (IEEE VCIP 2013)*. 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2013. ISBN 978-1-4799-0288-0. IEEE International Conference on Visual Communications and Image Processing (VCIP), Kuching, MALAYSIA, NOV 17-20, 2013. Citado 4 vezes nas páginas 37, 39, 41, and 66.

ITTI, L.; KOCH, C.; NIEBUR, E. A model of saliency-based visual attention for rapid scene analysis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, v. 20, n. 11, p. 1254–1259, NOV 1998. ISSN 0162-8828. Citado na página 34.

KARIMI, M. et al. Pyramidal modeling of geometric distortions for retargeted image quality evaluation. *MULTIMEDIA TOOLS AND APPLICATIONS*, v. 77, n. 11, p. 13799–13820, JUN 2018. ISSN 1380-7501. Citado 3 vezes nas páginas 38, 41, and 66.

KARNI, Z.; FREEDMAN, D.; GOTSMAN, C. Energy-based image deformation. *COMPUTER GRAPHICS FORUM*, v. 28, n. 5, SI, p. 1257–1268, JUL 2009. ISSN 0167-7055. 7th Eurographics Symposium on Geometry Processing (SGP), Berlin, GERMANY, JUL 15-17, 2009. Citado na página 34.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. Disponível em: <<https://api.semanticscholar.org/CorpusID:6628106>>. Citado na página 45.

KINGMA, D. P.; WELLING, M. An introduction to variational autoencoders. *FOUNDATIONS AND TRENDS IN MACHINE LEARNING*, v. 12, n. 4, p. 4–89, 2019. ISSN 1935-8237. Citado na página 49.

KRÄHENBÜHL, P. et al. A system for retargeting of streaming video. *ACM Trans. Graph.*, Association for Computing Machinery, New York, NY, USA, v. 28, n. 5, p. 1–10, dec 2009. ISSN 0730-0301. Disponível em: <<https://doi.org/10.1145/1618452.1618472>>. Citado na página 34.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *NATURE*, v. 521, n. 7553, p. 436–444, MAY 28 2015. ISSN 0028-0836. Citado na página 43.

LECUN, Y. et al. Backpropagation applied to handwritten zip code recognition. *NEURAL COMPUTATION*, v. 1, n. 4, p. 541–551, WIN 1989. ISSN 0899-7667. Citado na página 45.

LI, L. et al. Quality evaluation for image retargeting with instance semantics. *IEEE TRANSACTIONS ON MULTIMEDIA*, v. 23, p. 2757–2769, 2021. ISSN 1520-9210. Citado 3 vezes nas páginas 37, 41, and 66.

LI, P.; PEI, Y.; LI, J. A comprehensive survey on design and application of autoencoder in deep learning. *Applied Soft Computing*, v. 138, p. 110176, 2023. ISSN 1568-4946. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1568494623001941>>. Citado 3 vezes nas páginas 30, 48, and 58.

LI, Y.; GUO, L.; JIN, L. A content-aware image retargeting quality assessment method using foreground and global measurement. *IEEE ACCESS*, v. 7, p. 91912–91923, 2019. ISSN 2169-3536. Citado 3 vezes nas páginas 38, 41, and 66.

LIN, J. et al. Objective quality assessment for image retargeting based on hybrid distortion pooled model. In: *2015 SEVENTH INTERNATIONAL WORKSHOP ON QUALITY OF MULTIMEDIA EXPERIENCE (QOMEX)*. 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2015. (International Workshop on Quality of Multimedia Experience). ISBN 978-1-4799-8958-4. ISSN 2372-7179. 7th International Workshop on Quality of Multimedia Experience (QoMEX), Messinia, GREECE, MAY 26-29, 2015. Citado 3 vezes nas páginas 39, 41, and 66.

LIU, C. et al. Sift flow: Dense correspondence across different scenes. In: FORSYTH, D.; TORR, P.; ZISSERMAN, A. (Ed.). *COMPUTER VISION - ECCV 2008, PT III, PROCEEDINGS*. HEIDELBERGER PLATZ 3, D-14197 BERLIN, GERMANY: SPRINGER-VERLAG BERLIN, 2008. (Lecture Notes in Computer Science, 3), p. 28–42. ISBN 978-3-540-88689-1. ISSN 0302-9743. 10th European Conference on Computer Vision (ECCV 2008), Marseille, FRANCE, OCT 12-18, 2008. Citado 3 vezes nas páginas 39, 41, and 66.

LIU, Y.-J. et al. Image retargeting quality assessment. *COMPUTER GRAPHICS FORUM*, v. 30, n. 2, p. 583–592, 2011. ISSN 0167-7055. Citado 4 vezes nas páginas 39, 40, 41, and 66.

MA, L. et al. Retargeted image quality assessment: Current progresses and future trends. In: _____. *Visual Signal Quality Assessment: Quality of Experience (QoE)*. Cham: Springer International Publishing, 2015. p. 213–242. ISBN 978-3-319-10368-6. Disponível em: <https://doi.org/10.1007/978-3-319-10368-6_8>. Citado na página 30.

MA, L. et al. Image retargeting quality assessment: A study of subjective scores and objective metrics. *IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING*, v. 6, n. 6, p. 626–639, OCT 2012. ISSN 1932-4553. Citado 3 vezes nas páginas 30, 37, and 40.

MANJUNATH, B. et al. Color and texture descriptors. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, v. 11, n. 6, p. 703–715, JUN 2001. ISSN 1051-8215. Citado 3 vezes nas páginas 39, 41, and 66.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, v. 5, n. 4, p. 115–133, 1943. Citado na página 43.

NG, A. et al. Sparse autoencoder. *CS294A Lecture notes*, v. 72, n. 2011, p. 1–19, 2011. Citado na página 49.

NIU, Y. et al. Image retargeting quality assessment based on registration confidence measure and noticeability-based pooling. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, v. 31, n. 3, p. 972–985, MAR 2021. ISSN 1051-8215. Citado 3 vezes nas páginas 38, 41, and 66.

PELE, O.; WERMAN, M. Fast and robust earth mover's distances. In: IEEE; IEEE COMP SOC. *2009 IEEE 12TH INTERNATIONAL CONFERENCE ON COMPUTER*

VISION (ICCV). 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2009. (IEEE International Conference on Computer Vision), p. 460–467. ISBN 978-1-4244-4419-9. ISSN 1550-5499. 12th IEEE International Conference on Computer Vision, Kyoto, JAPAN, SEP 29-OCT 02, 2009. Citado 3 vezes nas páginas 40, 41, and 66.

PRITCH, Y.; KAV-VENAKI, E.; PELEG, S. Shift-map image editing. In: IEEE; IEEE COMP SOC. *2009 IEEE 12TH INTERNATIONAL CONFERENCE ON COMPUTER VISION (ICCV)*. 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2009. (IEEE International Conference on Computer Vision), p. 151–158. ISBN 978-1-4244-4419-9. ISSN 1550-5499. 12th IEEE International Conference on Computer Vision, Kyoto, JAPAN, SEP 29-OCT 02, 2009. Citado 3 vezes nas páginas 17, 34, and 35.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: NAVAB, N. et al. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015. p. 234–241. ISBN 978-3-319-24574-4. Citado 2 vezes nas páginas 50 and 51.

RUBINSTEIN, M. et al. A comparative study of image retargeting. *ACM Trans. Graph.*, Association for Computing Machinery, New York, NY, USA, v. 29, n. 6, dec 2010. ISSN 0730-0301. Disponível em: <<https://doi.org/10.1145/1882261.1866186>>. Citado 3 vezes nas páginas 29, 30, and 37.

RUBINSTEIN, M.; SHAMIR, A.; AVIDAN, S. Multi-operator media retargeting. *ACM Trans. Graph.*, Association for Computing Machinery, New York, NY, USA, v. 28, n. 3, jul 2009. ISSN 0730-0301. Disponível em: <<https://doi.org/10.1145/1531326.1531329>>. Citado na página 34.

RUSSELL, S. J.; NORVIG, P. *Artificial Intelligence: A Modern Approach (4th Edition)*. Pearson, 2020. ISBN 9781292401133. Disponível em: <<http://aima.cs.berkeley.edu/>>. Citado 3 vezes nas páginas 43, 44, and 45.

SHAMIR, A.; SORKINE, O. Visual media retargeting. In: *ACM SIGGRAPH ASIA 2009 Courses*. New York, NY, USA: Association for Computing Machinery, 2009. (SIGGRAPH ASIA '09). ISBN 9781450379311. Disponível em: <<https://doi.org/10.1145/1665817.1665828>>. Citado 2 vezes nas páginas 29 and 33.

SIMAKOV, D. et al. Summarizing visual data using bidirectional similarity. In: IEEE COMP SOC. *2008 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, VOLS 1-12*. 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2008. (IEEE Conference on Computer Vision and Pattern Recognition), p. 3887+. ISBN 978-1-4244-2242-5. ISSN 1063-6919. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, JUN 23-28, 2008. Citado 3 vezes nas páginas 39, 41, and 66.

SUGANUMA, M.; OZAY, M.; OKATANI, T. Exploiting the potential of standard convolutional autoencoders for image restoration by evolutionary search. In: DY, J.; KRAUSE, A. (Ed.). *INTERNATIONAL CONFERENCE ON MACHINE LEARNING, VOL 80*. 1269 LAW ST, SAN DIEGO, CA, UNITED STATES: JMLR-JOURNAL MACHINE LEARNING RESEARCH, 2018. (Proceedings of Machine Learning Research, v. 80). ISSN 2640-3498. 35th International Conference on Machine Learning (ICML), Stockholm, SWEDEN, JUL 10-15, 2018. Citado 2 vezes nas páginas 58 and 59.

VAHDAT, A.; KAUTZ, J. Nvae: A deep hierarchical variational autoencoder. In: LAROCHELLE, H. et al. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2020. v. 33, p. 19667–19679. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2020/file/e3b21256183cf7c2c7a66be163579d37-Paper.pdf>. Citado na página 57.

VAQUERO, D. et al. A survey of image retargeting techniques. In: TESCHER, A. (Ed.). *APPLICATIONS OF DIGITAL IMAGE PROCESSING XXXIII*. 1000 20TH ST, PO BOX 10, BELLINGHAM, WA 98227-0010 USA: SPIE-INT SOC OPTICAL ENGINEERING, 2010. (Proceedings of SPIE, v. 7798). ISBN 978-0-8194-8294-5. ISSN 0277-786X. Conference on Applications of Digital Image Processing XXXIII, San Diego, CA, AUG 02-04, 2010. Citado na página 29.

VASWANI, A. et al. Attention is all you need. In: GUYON, I. et al. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>. Citado 2 vezes nas páginas 50 and 51.

VINCENT, P. et al. Extracting and composing robust features with denoising autoencoders. In: *Proceedings of the 25th International Conference on Machine Learning*. New York, NY, USA: Association for Computing Machinery, 2008. (ICML '08), p. 1096–1103. ISBN 9781605582054. Disponível em: <<https://doi.org/10.1145/1390156.1390294>>. Citado na página 49.

WANG, Y.-S. et al. Optimized scale-and-stretch for image resizing. *ACM Trans. Graph.*, Association for Computing Machinery, New York, NY, USA, v. 27, n. 5, dec 2008. ISSN 0730-0301. Disponível em: <<https://doi.org/10.1145/1409060.1409071>>. Citado na página 34.

WANG, Z. et al. Image quality assessment: From error visibility to structural similarity. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 13, n. 4, p. 600–612, APR 2004. ISSN 1057-7149. Citado 2 vezes nas páginas 29 and 35.

WOLF, L.; GUTTMANN, M.; COHEN-OR, D. Non-homogeneous content-driven video-retargeting. In: IEEE. *2007 IEEE 11TH INTERNATIONAL CONFERENCE ON COMPUTER VISION, VOLS 1-6*. 345 E 47TH ST, NEW YORK, NY 10017 USA: IEEE, 2007. (IEEE International Conference on Computer Vision), p. 1418–1423. ISBN 978-1-4244-1630-1. ISSN 1550-5499. 11th IEEE International Conference on Computer Vision, Rio de Janeiro, BRAZIL, OCT 14-21, 2007. Citado na página 34.

YAN, B. et al. Learning quality assessment of retargeted images. *Signal Processing: Image Communication*, v. 56, p. 12–19, 2017. ISSN 0923-5965. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0923596517300711>>. Citado 5 vezes nas páginas 30, 37, 41, 65, and 66.

ZHANG, Y. et al. Backward registration-based aspect ratio similarity for image retargeting quality assessment. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, v. 25, n. 9, p. 4286–4297, SEP 2016. ISSN 1057-7149. Citado 4 vezes nas páginas 38, 40, 41, and 66.

ZHANG, Y. et al. Multiple-level feature-based measure for retargeted image quality. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, v. 27, n. 1, p. 451–463, JAN 2018. ISSN 1057-7149. Citado 3 vezes nas páginas 38, 41, and 66.

Appendix

APPENDIX A – Fusion combination strategies from AEIMPS

For the sake of simplicity, consider both $\mathbf{X}, \mathbf{Y}$ are tensors of rank-3 (rows, columns, and channels) such as $\mathbf{X} \in \mathbb{R}^{M \times N \times C}$ and $\mathbf{Y} \in \mathbb{R}^{M' \times N' \times C}$. Next, we describe the basis information and their combination.

$$\text{cmae}(\mathbf{X}, \mathbf{Y}) = 1 - \frac{\|\bar{\mathbf{X}} - \bar{\mathbf{Y}}\|_1}{M'' \times N'' \times C}. \quad (\text{A.1})$$

$$\text{fnorm}(\mathbf{X}) = \|\bar{\mathbf{X}}\|_{\mathcal{F}}. \quad (\text{A.2})$$

$$\text{cnorm}(\mathbf{X}) = \frac{1}{C} \sum_{c=1}^C \text{fnorm}(\bar{\mathbf{X}}_{*,*,c}). \quad (\text{A.3})$$

$$\rho(\mathbf{X}, \mathbf{Y}) = \frac{\min(M, M') \min(N, N')}{\max(M, M') \max(N, N')}. \quad (\text{A.4})$$

Also, the $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}} \in \mathbb{R}^{M'' \times N'' \times C}$ stand for their scaled version between 0 and 1, i.e., $0 \leq X_{*,*,*}, Y_{*,*,*} \leq 1$. Moreover, note they are also resized to have the same dimensions.

Regarding the derived fusion strategies, we detail them as follows:

- a) **FUSION 01.** The Compl. of Mean Absolute Error score between the image and its retargeted reconstruction from the AE, i.e., $\text{cmae}(\mathbf{I}, d(e(\mathbf{J})))$;
- b) **FUSION 02.** Weighted Compl. of Mean Absolute Error by retargeting factor between the image and its retargeted reconstruction, i.e., $\rho(\mathbf{I}, \mathbf{J}) \cdot \text{cmae}(\mathbf{I}, d(e(\mathbf{J})))$;
- c) **FUSION 03.** Weighted Compl. of Mean Absolute Error by the inverse of Retargeting factor between the image and its retargeted reconstruction, i.e. $\rho(\mathbf{I}, \mathbf{J})^{-1} \text{cmae}(\mathbf{I}, d(e(\mathbf{J})))$;
- d) **FUSION 04.** The Compl. of Mean Absolute Error score between the image and its reconstruction from the AE to the power of the Retargeting factor, i.e., $\text{cmae}(\mathbf{I}, d(e(\mathbf{J})))^{\rho(\mathbf{I}, \mathbf{J})}$;
- e) **FUSION 05.** Retargeting factor to the power of the Compl. of Mean Absolute Error score between the image and its reconstruction from the AE, i.e., $\rho(\mathbf{I}, \mathbf{J})^{\text{cmae}(\mathbf{I}, d(e(\mathbf{J})))}$;
- f) **FUSION 06 up to 09.** Linear combination (using α, β) between the Compl. of Mean Absolute Error score between the image and its reconstruction from the AE and Retargeting factor, i.e., $\alpha \cdot \text{cmae}(\mathbf{I}, d(e(\mathbf{J}))) + \beta \cdot \rho(\mathbf{I}, \mathbf{J})$. Fusion 6 with (0, 5, 0.6), Fusion 7 with (0.8, 0.2), Fusion 8 with (0.2, 0.8), and Fusion 9 with (0.1, 0.9);

- g) **FUSION 10.** The Compl. of Mean Absolute Error score between the retargeted and its retargeted reconstruction from the AE, i.e., $\text{cmae}(\mathbf{J}, d(e(\mathbf{J})))$.
- h) **FUSION 11.** Average norm by channel between the difference of original and retargeted image codes, i.e., $\text{cnorm}(e(\mathbf{I}) - e(\mathbf{J}))$.
- i) **FUSION 12.** Weighted Average norm by channel between the difference of original and retargeted image codes by the Retargeting factor, i.e., $\rho(\mathbf{I}, \mathbf{J}) \cdot \text{cnorm}(e(\mathbf{I}) - e(\mathbf{J}))$.
- j) **FUSION 13.** Norm of the retargeting code, i.e., $\text{norm}(e(\mathbf{J}))$.

In Table 4, we show the results from each fusion strategy.

Table 4 – Fusion strategies results.

IRQA	LLC	SRCC	RMSE	OR
Fusion 01	0.3404	0.3175	12.6944	0.0
Fusion 02	0.5288	0.4544	11.4587	0.0
Fusion 03	0.4313	0.3598	12.1809	0.0
Fusion 04	0.0436	0.0563	13.4880	0.0
Fusion 05	0.4963	0.3584	11.7205	0.0
Fusion 06	0.5268	0.4546	11.4756	0.0
Fusion 07	0.4476	0.4225	12.0730	0.0
Fusion 08	0.5356	0.4557	11.4007	0.0
Fusion 09	0.5333	0.4552	11.4211	0.0
Fusion 10	0.1390	0.1190	13.3699	0.0
Fusion 11	0.3314	0.3209	12.7382	0.0
Fusion 12	0.1250	0.1154	13.3972	0.0
Fusion 13	0.3774	0.4216	12.5023	0.0