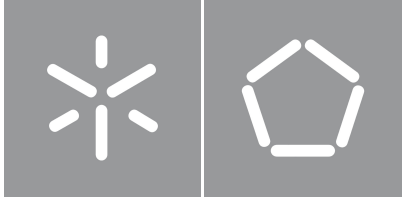
University of Minho
School of Engineering

Lucas Silva Carvalho

Deep Learning Super Resolution: Exploring Loss Functions, Metrics, and Datasets

january 2025



University of Minho
School of Engineering

Lucas Silva Carvalho

Deep Learning Super Resolution: Exploring Loss Functions, Metrics, and Datasets

Masters Dissertation
Master's in Informatics Engineering

Dissertation supervised by
António Ramires

january 2025

Copyright and Terms of Use for Third Party Work

Acknowledgements

After five years, I have finally reached the end of this journey, and it would not have been possible without the support of numerous people to whom i wish to express my gratitude.

First and foremost, i want to thank my parents for their unwavering support throughout the years. Without your dedication and encouragement, i would not be where i am today. To my girlfriend, your encouragement kept me focused and motivated during challenging moments.

I must express my sincere thanks to Professor António Ramires for his constant guidance, knowledge, and encouragement to improve, which were essential in helping me complete this work with the finest quality. Your commitment and insights made a significant difference.

A special thanks goes to my friends Duarte, Tiago, Pedro, and Manuel, who were also working on their theses and together brought me countless moments of enjoyment and happiness. Thank you guys! We made it!

Statement of Integrity

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, Braga, january 2025

Lucas Silva Carvalho

Abstract

Single Image Super Resolution (SISR) is an active area of research within the deep learning community, with the objective of enhancing the resolution of low-resolution (LR) images to generate high-resolution (HR) counterparts. The primary goal is to recover finer details and improve image quality, making it valuable for a wide range of applications that benefit from more detailed images. The network architecture is usually the main focus, however, there are other aspects that also influence the final result which would benefit from further research.

We begin by examining different loss functions employed in SISR models, such as pixel loss and perceptual loss, analyzing how they affect the performance of the final result, and using standard metrics to assess these methods. The evaluation of these different metrics on the quality and accuracy of the images is critically examined using HR generated samples. Furthermore, a subjective user study was conducted on these samples, serving not only to evaluate the alignment between the objective metrics and human perception but also to reinforce and validate the conclusions drawn from the visual analysis.

Furthermore, we explore the significance of the training datasets, considering how factors such as dataset specificity, size, and the method used to generate the LR samples (bicubic downscaling and real degradations) influence the effectiveness of SISR models. Both objective metrics and subjective assessments of HR samples are used to gauge the effectiveness of different training datasets, providing a balanced perspective on model performance.

Finally, two distinct SISR approaches are compared: the supervised approach, where models are pre-trained on large-scale datasets with paired LR-HR images, allowing them to learn the direct mapping between both counterparts; and the unsupervised approach, which operates without the need for paired datasets, generating HR images from a single LR image without prior training.

Keywords Super-resolution, evaluation metrics, unsupervised, supervised, deep leaning, datasets, loss functions.

Resumo

Single Image Super Resolution (SISR) é uma área de investigação ativa na comunidade de aprendizagem profunda, com o objetivo de melhorar a resolução de imagens de baixa resolução (BR) para gerar as suas correspondências de alta resolução (AR). O objetivo principal é recuperar detalhes mais refinados e melhorar a qualidade da imagem, tornando-a valiosa para uma ampla gama de aplicações que beneficiam de imagens mais detalhadas. A arquitetura da rede é geralmente o foco principal; no entanto, há outros aspetos que também influenciam o resultado final e que beneficiariam de uma investigação aprofundada.

Começamos por examinar diferentes funções de perda utilizadas em modelos SISR, como a perda de píxeis e a perda perceptual, analisando como estas afetam o desempenho do resultado final, através de métricas padrão para avaliar estes métodos. A avaliação dessas diferentes métricas em termos de qualidade e precisão das imagens é criticamente analisada com recurso a amostras AR geradas. Além disso, foi realizado um estudo subjetivo com usuários sobre estas amostras, servindo não apenas para avaliar a concordância entre as métricas objetivas e a percepção humana, mas também para reforçar e validar as conclusões tiradas das análises visuais.

Adicionalmente, exploramos a importância dos conjuntos de dados de treino, considerando de que forma fatores como a especificidade, o tamanho e o método utilizado para gerar as amostras BR (redução bicúbica e degradações reais) influenciam a eficácia dos modelos SISR. Tanto métricas objetivas como avaliações subjetivas das amostras AR são utilizadas para aferir a eficácia de diferentes conjuntos de dados de treino, proporcionando uma perspetiva equilibrada sobre o desempenho dos modelos.

Por fim, são comparadas duas abordagens distintas de SISR: a abordagem supervisionada, onde os modelos são pré-treinados em conjuntos de dados de larga escala com imagens BR-AR emparelhadas, permitindo que aprendam o mapeamento direto entre ambas as amostras; e a abordagem não supervisionada, que opera sem a necessidade de conjuntos de dados emparelhados, gerando imagens AR a partir de uma única imagem BR sem treinamento prévio.

Palavras-chave Super-resolução, métricas de avaliação, não-supervisionado, supervisionado, aprendizagem profunda, conjuntos de dados, funções de perda.

Contents

List of Figures

List of Tables

Part I

Introductory material

Chapter 1

Introduction

The demand for enhanced image resolution is present in areas such as medical imaging, satellite imagery and even gaming. The goal behind Single Image Super Resolution (SISR) is to reconstruct a high-resolution (HR) image from a low-resolution (LR) observation, where it aims to recover the lost high-frequency details required for obtaining natural and/or informative HR images.

Prior to the entrance of deep learning methods in this area, interpolation methods like bicubic or Lanczos, often failed to produce convincing results, with results getting degraded as the upscaling factor goes up Yang et al. [2019]. These methods tend to generate overly smooth images, lacking the high frequency details present in the original HR scenes. In recent years, the deep learning community has developed methods to address the limitations of conventional SISR techniques, with Convolutional Neural Networks (CNNs) achieving some successes in capturing high frequency information required to create HR images Dong et al. [2014].

1.1 Motivation

There are several relevant aspects that influence the effectiveness of SR deep learning models. Loss functions guide the optimization process, and their choice significantly impacts the quality of the generated HR images. The training dataset can also have a major role in the outcome, depending on the variety, number of images and the process to obtain the LR samples. However, how can we benefit from these factors to improve the quality of the reconstructed images still remains a question.

Additionally, the emergence of unsupervised learning approaches in SISR [Shocher et al., 2018] introduce another layer of complexity to the area. While supervised methods rely on large paired datasets, unsupervised approaches leverage image-specific information without requiring prior training. Understanding the strengths and limitations of each method is crucial for further improving SISR techniques.

Furthermore, a comprehensive analysis of the metrics used to evaluate SR results is essential to

understand how they correlate with subjective assessments of image quality [Ledig et al., 2017].

1.2 Objectives

The primary objectives of this study are to analyze various factors that influence the performance of SISR models mentioned on the previous section:

- **Examine the impact of different loss functions:** We aim to explore how different loss functions, pixel-loss and perceptual-loss, affect the outcome of SISR models.
- **Analyze the effect of training datasets:** We aim to investigate how the specificity, cardinality and types of LR images affect the performance of SISR models by using different levels of tailoring, as well as exploring a different process to simulate real LR images.
- **Compare supervised and unsupervised approaches:** We contrast supervised SISR with an unsupervised method, aiming to address the strengths and limitations of each approach in terms of reconstruction quality in different conditions and training effort.
- **Explore evaluation metrics:** The study seeks to assess different pixel-based and content-based metrics, investigating how well these metrics correlate with human perception to evaluate the quality of generated HR images.

1.3 Document Structure

This document is organized into several main chapters, each addressing distinct aspects of the research. We start by providing a comprehensive overview of the main aspects and methodologies in this field (Chapter 2).

In Chapter 3 we detail the methodologies and contributions of the research, discussing the datasets used and the network details of the supervised and unsupervised approaches explored.

The outcomes of the proposed methodologies of both supervised and unsupervised methods, are presented and evaluated in Chapter 4.

Finally, Chapter 5 summarizes the findings of the research and outlines potential future directions.

In addition to the main chapters, the Appendix Section A presents our article that covers a big part of the content discussed throughout this dissertation, published in The International Conference on Graphics and Interaction (ICGI) 2024.

Chapter 2

State of the Art

This chapter provides a comprehensive review of key concepts and methodologies in the field of Super-Resolution (SR). We begin with an introduction to SR in Section 2.1, discussing the significance of this topic as well as the available techniques to perform it.

The most popular datasets used for both training and evaluating SR models are presented in Section 2.2.

Furthermore, we explore upsampling methods (Section 2.3), frameworks in which SR learning based models operate (Section 2.4), some popular network architectures and strategies among the SR comunity (Section 2.5), followed by the different types of loss functions (Section 2.6).

Section 2.7 covers various image quality assessment methods, explaining widely used metrics, along with other alternative methods.

Finally, a comparison of different model's results is presented in Section 2.8.

2.1 Super Resolution

SR is known as the task of creating high-resolution (HR) images from its corresponding low-resolution (LR) version. There are two types of SR, according to the number of input LR images, the most popular is the single image super-resolution (SISR) which aims to recreate a HR version from a single LR image. On the other hand, the multi-image super-resolution (MISR) uses multiple LR images, often from different perspectives or time instances, to create a single HR image.

Since an HR image has more valuable details, it can be used in many areas, such as security, satellite, medical imaging and gaming.

Considering a HR image which is used to create a LR image, the LR image y can be modeled as the output of the process described in Figure 1 and the following Equation 2.1.

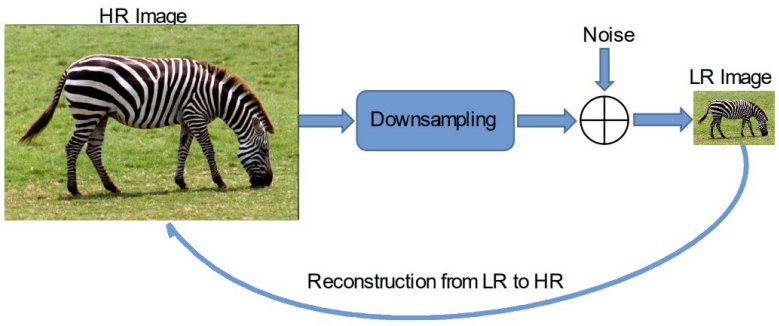


Figure 1: Sketch of the overall framework of SISR.

$$y = D(x; \delta) \tag{2.1}$$

In Equation 2.1, *D* denotes a degradation mapping function, x is the corresponding HR image and δ are the parameters of the degradation process (the scaling factor and noise).

The goal then becomes to recover a HR approximation of x, $\hat{x}$, by minimizing the loss function between x and the SR image $\hat{x}$. Solving this problem is a challenging task due to the amount of lost information in the LR image.

To date, we have three main categories of algorithms for SISR:

- **Interpolation-based methods** like bicubic interpolation [Keys, 1981] are quick and straightforward but often lack accuracy, especially in generating sharp details.
- **Reconstruction-based** methods [Sun et al., 2008, Yan et al., 2015] use prior knowledge to restrict solution spaces, yielding sharp details. However, as the scale factor increases, their performance tends to degrade, and the computational demand is usually high.
- **Learning-based SISR** methods have been receiving more and more attention due to their fast computation and high performance Yang et al. [2019]. These methods usually use machine learning algorithms to analyze relationships between the LR and its corresponding HR counterpart from substantial training examples. Recently, deep learning based SISR algorithms have demonstrated great superiority to reconstruction based and other learning-based methods.

2.2 Datasets for Super Resolution

When training a SR network, the datasets used are an important part of the process. Today there are a variety of datasets available for image SR (for either training or testing), which greatly differ in image amounts, quality, resolution, and diversity. The datasets only provide HR images and the LR images are usually obtained with interpolation methods [Dong et al., 2014, Shocher et al., 2018, Ledig et al., 2017, Niu et al., 2020]. In Table 1 are listed the most popular image datasets used by the SR community, specifically indicating the number of images, average resolution and category.

In order to compare the results with previous works, the most used evaluation datasets tend to remain the same (BSDS100, General-100, Set5, Set14, Urban100). Additionally, some researchers like Niu et al. [2020] also use other domain datasets like Manga109 or even competition datasets like PIRM, the perceptual image restoration and manipulation challenge in collaboration with the European Conference on Computer vision (ECCV). In what comes to the training datasets, the DIV2K is the most popular among the SR community [Lim et al., 2017, Niu et al., 2020, Wang et al., 2018b]. However, The optimal size of the training dataset remains a question, some researchers rely only on DIV2k while others combine it with the OutdoorScene and Flickr2K, like Haris et al. [2018] or use the ImageNet database [Deng et al., 2009] to build massive datasets [Ledig et al., 2017, Chen et al., 2021, Huang et al., 2017].

Dataset	N° images	Avg. Resolution	Category	Source
BSD100	100	(432; 370)	general	[Martin et al., 2001]
DIV2K	1000	(1972; 1437)	general	[Agustsson and Timofte, 2017]
General-100	100	(435; 381)	general	[Dong et al., 2016]
Manga109	109	(826; 1169)	manga volume	[Fujimoto et al., 2016]
OutdoorScene	10624	(553; 440)	general	[Wang et al., 2018a]
PIRM	200	(617; 482)	general	[Blau et al., 2018]
Set5	5	(313; 336)	general	[Bevilacqua et al., 2012]
Set14	14	(492; 446)	general	[Zeyde et al., 2012]
Urban100	100	(984; 797)	urban landscapes	[Huang et al., 2015]
Flickr2K	2650	(1970; 1434)	general	[Timofte et al., 2017]

Table 1: Benchmark datasets for SISR

2.3 Upsampling methods

The method used to perform upsampling plays a crucial role in SR. The continuous research in this area introduced deep learning-based upsampling layers that, unlike the traditional predefined mathematical functions, allow for more adaptive and flexible transformations, leading to their adoption as a dominant approach in modern SR techniques.

2.3.1 Interpolation-based

Image interpolation is widely used by image-related applications and is known as the task of resizing digital images. Three of the most used interpolation-based methods [Rajarapollu and Mankar, 2017] are the nearest-neighbor, bilinear and bicubic interpolation. Since these methods are very straightforward and easy to implement, some of them are still used in deep learning SR models, especially the bicubic interpolation [Dong et al., 2014].

- **Nearest-neighbor interpolation** is one of the simplest and intuitive algorithms for image scaling. It quickly chooses the value of the closest pixel to fill in each new position during interpolation, without considering any other pixels. Despite its fast operation, this method often creates pixelated and poor-quality results due to its simplistic approach.

- **Bilinear interpolation** computes new pixel values based on linear interpolations along each axis of the image. This means that interpolation is done horizontally and vertically, resulting in a quadratic interpolation with a receptive field of size 2x2. Unlike nearest-neighbor, bilinear interpolation uses the intensity of the neighbor pixels, taking into account the contribution of each. This results in smoother transitions and consequentially, in a higher quality image.

- **Bicubic interpolation** extends the idea of bilinear interpolation. It performs cubic interpolation along the horizontal and vertical axes, considering a larger area of pixels. This allows the results to be smoother and with fewer artifacts.

2.3.2 Learning-based

In order to overcome the shortcomings of interpolation-based methods and learn upsampling in an end-to-end manner, learning-based upsampling methods such as the sub-pixel layer and the deconvolution layer were introduced into the SR field:

- **sub-pixel layer** [Shi et al., 2016a] performs the upsampling by first employing convolutional operations that create extra channels expanding the depth of the data and providing more information to work with. If the input image is $h*w*c$ (height, width, channels) and r the upscaling factor, this process expands it to $h * w * (c * r^2)$. Finally, to generate the final upscaled image, a reshaping operation is performed to get an output with size $rh * rw * c$, this process is shown in Figure 2a for an scale factor of 2. Having a large receptive field provides more contextual information to help generate more realistic details, which makes this layer widely used in SR.

- **Deconvolution layer**, also known as transposed convolution layer [Zeiler et al., 2010], reverses the convolutional process by predicting a HR image from LR feature maps. This layer starts by increasing the size of the LR image by adding empty spaces (filled with zeros), after this expansion, a convolution operation using a specific kernel size is applied. Image 2b illustrates this process with a scaling factor of 2 and a convolution kernel of 3 × 3.

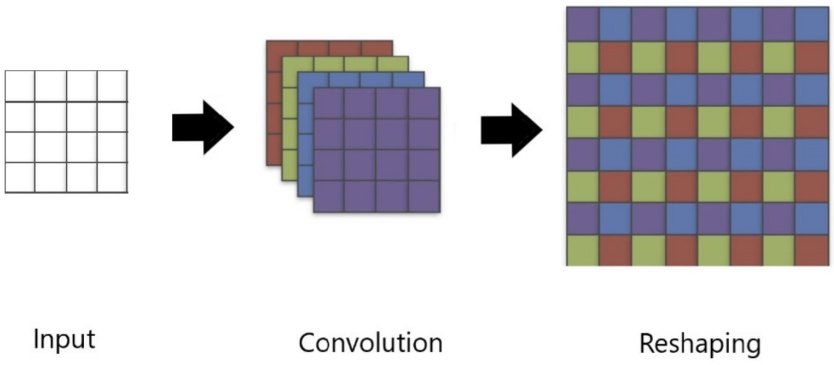
- **Meta upscaling** module [Hu et al., 2019] aims at addressing the limitation of having to predefine the scaling factor for resolution enhancement allowing the use of various arbitrary scaling factors to generate a high-resolution image. This module operates by considering the relationship between the LR and HR feature maps. Each position in the desired HR image is mapped to a small block in the LR feature maps using an arbitrary patch size, then the convolutional weights are generated to transform these LR patches into a new set of channels corresponding to the output image.

  Although not as extensively explored as some other methods, meta upscaling is a very interesting approach for its flexibility in handling arbitrary scaling factors, but in situations requiring high magnifications, there might be challenges with the model's stability, as it predicts convolution weights for each pixel.

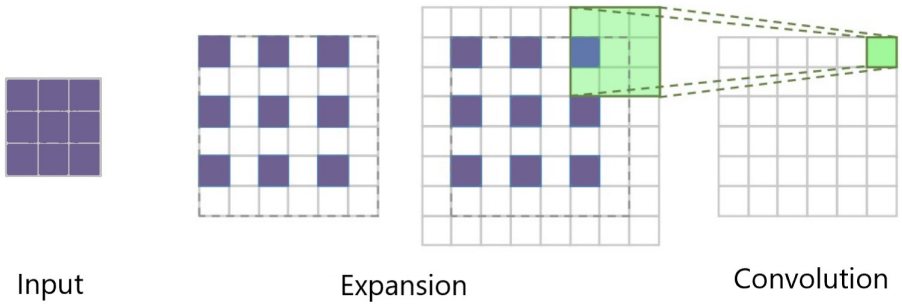
2.4 Super Resolution Frameworks

Existing models for super-resolution exhibit diverse architectures but can generally be categorized into four main frameworks (Figure 3) based on the upsampling operations used and their placement within the model. The distinct strategies impact the final output quality and computational efficiency of the models:

- In the **pre-upsampling** SR framework, the approach involves learning mapping functions that directly transform a LR image into a HR image. This process begins with the LR image being



(a) Sub-pixel layer



(b) Deconvolution layer

Figure 2: Upsampling process in the sub-pixel and deconvolutional layers

converted to HR image (Figure 3a) using the classical interpolation-based upsampling methods, so the network's job is to refine the results, what leads to a reduction in learning difficulty. Using this concept the pre-upsampling-based SR framework was introduced by Dong et al. [2014], inspiring many other works, such as [Tai et al., 2017b] and [Kim et al., 2016].

However, the predefined upsampling often introduces side effects (e.g., noise amplification and blurring) Wang et al. [2021], and since most operations are performed in high-dimensional space, the time and space demands significantly impact the overall performance compared to other frameworks [Shi et al., 2016b].

- In **post-upsampling**, with the objective of improving the computational efficiency and make full use of deep learning technology to increase resolution, researchers [He et al., 2016, Lim et al., 2017] perform most computation in low-dimensional space by replacing the predefined upsampling with end-to-end learnable layers integrated at the end of the models (Figure 3b).

  The low-dimensional space computation advantage, as made Post-upsampling become one of the most mainstream frameworks [He et al., 2016, Lim et al., 2017, Wang et al., 2018b].

- The post-upsampling SR framework significantly reduced computational costs, yet it performed upsampling in a single step, making learning challenging for larger scaling and requires individual training for each scaling factor. To overcome these drawbacks, works like the Laplacian pyramid SR network (LapSRN) [Lai et al., 2017] adopted a **progressive upsampling** approach (Figure 3c), employing a cascade of convolutional neural networks (CNNs) to progressively reconstruct higher-resolution images, so at each stage, images are upsampled and refined using CNNs.

  Within this framework, models simplify complex tasks, making the learning process easier, and capable of handling multi-scale SR. Nonetheless, these models face challenges such as the complicated model design for multiple stages, increased Resource Demands and training instability due to error propagation on the first scales [Wang et al., 2021].

- In order to enhance the understanding of the interdependent relationship between LR and HR image pairs, **iterative up-and-down sampling** refines the image using recursive back-propagation, by continuously measuring the error and refining the model based on the reconstruction error (Figure 3d).

  Specifically, Haris et al. [2018] exploit iterative up-and-down sampling layers and propose DBPN (Deep Back-Projection Networks), which connects upsampling and downsampling layers alternately and reconstructs the final HR result using all of the intermediately reconstructions.

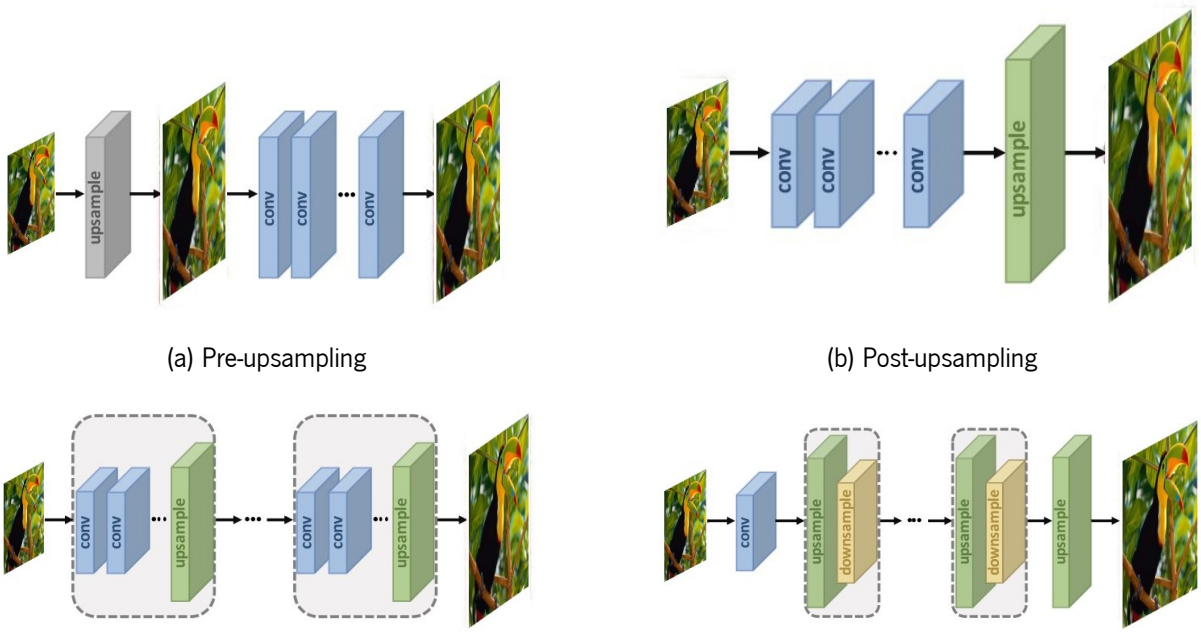
  The repeated iterations and refinement stages increase the processing time and resource demands, as well as memory requirements, especially if the iterative process retains or accumulates intermediate representations.

2.5 SR network architectures and strategies

The network designs and architecture advancements are crucial in deep learning, for SR specifically, researchers have proposed several design implications along with the SR framework for designing the overall SR network.

The pioneer for SR with deep learning was proposed by Dong et al. [2014] as SRCNN, a pre-upsampling (bicubic HR is the input) three layer CNN that approximates the complex mapping between the LR and HR spaces. This method immediately showed vast superiority over the traditional ones, and motivated the blooming of CNN based SR methods.

Here, some of the fundamental network designs and strategies for SISR are presented.



(a) Pre-upsampling

(b) Post-upsampling

(c) Progressive upsampling

(d) Iterative up-and-down sampling

Figure 3: Representations of the SR frameworks

Residual learning was first proposed by He et al. [2016] (ResNet) aiming at facilitating the training of very deep networks. In the context of SR, LR and HR images share much information, so, rather than learning the complete mapping, the network focuses on modeling the difference or residual image between these LR and HR images.

The original ResNet was designed for classification, inspiring researchers to apply this concept in SR, such as Ledig et al. [2017] in Super Resolution Residual Network (SRResNet), which is composed of 16 residual blocks (a residual block has become the basic unit in these network structure consisting of two 3×3 convolutional layers, two batch normalization layers, and one ReLU activation function in between). The network's results were further improved by Lim et al. [2017], a similar, but much deeper network with the cost of a higher computational demand, the Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR).

Residual learning is one of the most used architectures for SR, not only facilitating the training of very deep networks but also contributing to the design of lightweight models that still achieve good results [Gendy et al., 2023].

In **recursive learning** the strategy is based on the same module being updated on a iterative way, minimizing the parameters.

The Deeply-recursive Convolutional Network, proposed by Kim et al. [2016] is one of the most used

recursive networks for SR, it uses a single convolution layer that reaches up to a 41 × 41 repetitive field without requiring additional parameters.

Despite reducing the parameters, recursive learning networks require higher computational power and too many stacked layers may cause the problem of vanishing/exploding gradient (when the gradients of the loss function become extremely small or excessively large), what makes it often used in combination with other strategies, such as residual learning. For example, Tai et al. [2017a] in the Deep Recursive Residual Network and Tai et al. [2017b] in the Memory Network for Image Restoration, both used residual blocks as part of the recursive module.

In **curriculum learning**, the approach involves gradually increasing the complexity of the learning task over time. This method proves beneficial for tasks involving sequence prediction or sequential decision-making, as it helps reduce training difficulty. For SISR, which faces inherent challenges like handling large scaling factors or noise, curriculum learning serves as an effective strategy. In LapSRN, curriculum learning is used by Lai et al. [2017] along with the progressive-upsampling framework to systematically reconstruct the sub-band residuals of high-resolution images in a progressive way. This strategy was also used by Li et al. [2019] to solve complex degradation tasks, where targets of different difficulties are ordered to learn progressively.

Dense connection, introduced in computer vision through DenseNet [Huang et al., 2017], unlike traditional architectures that transmit hierarchical features only to the final reconstruction layer, lies in each layer receiving input from all preceding layers. These connections form short paths between most layers address issues like vanishing or exploding gradients and helps flow of deep information across layers.

Inspired by DenseNet's success, Tong et al. [2017] integrated its dense connection mechanism into SISR introducing SRDenseNet. SRDenseNet not only employs dense connections at the layer level but extends this concept to the block level. Here, the output of each dense block is interconnected through dense connections, which allows the use of low-level and high-level features for the prediction. Later other works have been proposed, making use of dense connections with other strategies like multi-scale learning [Li et al., 2020] and residual learning [Mei et al., 2019].

Multi-scale learning aligns with iterative-upsampling framework, aiming to take advantage of the different characteristics that images may exhibit at different scales by making full use of these features to improve the model performance. For example, Li et al. [2018] proposed a multi-scale residual block to extract information in multiple scales and [Li et al., 2020] used dense connections to help the multi-scale information flow among different depths of the network.

Generative Adversarial Networks (GANs) consist of a generator and a discriminator. The discriminator is trained to judge whether an image is real or generated, the generator aims at fooling the discriminator rather than minimizing the distance to a specific image.

When proposing SRResNet, Ledig et al. [2017] also used it as a generator in SRGAN, where the discriminator is trained to distinguish SR images provided by the generator from the ground truth HR images.

Unsupervised SR is usually called "Weakly Supervised SR", where the model doesn't require HR and LR images pairs, instead, it uses two different datasets with uncorrelated images and uses a cycle-in-cycle approach to predict the mapping function of these two datasets (LR to HR and HR to LR images) using GANs [Yuan et al., 2018].

In recent years, a new technique was proposed to generate SR images, the **Zero-Shot Super Resolution** (ZSSR) [Shocher et al., 2018]. Instead of relying on a dataset with paired LR and HR images for training, ZSSR operates using a single LR image to directly train a deep learning model during the test time using image augmentation techniques, which involve generating multiple LR-HR pairs from the single LR image through various degradations. The final step is to train a SRCNN with the generated dataset, allowing the model to learn directly from the internal features and patterns within the image itself.

In addition to these architectures, numerous other designs have been proposed for SR. Many of these models tend to combine these foundational approaches with new, innovative techniques to further enhance performance and efficiency Gendy et al. [2023]. The continuous evolution and hybridization of these strategies highlight the dynamic and rapidly advancing field of super-resolution network architectures.

2.6 Loss functions

In the SISR task, the loss function role is to guide the iterative optimization process of the model by computing some error between the SR image and the HR image.

Pixel loss is the simplest and most popular type of loss function in SISR, it serves as a straightforward method to evaluate the disparity between two images at the pixel level, making them converge as close as possible.

These are usually known as L2 loss, corresponding to the mean square error (MSE) (Equation 2.6) and L1 loss also known as mean absolute error (MAE) represented by Equation 2.2.

$$MAE = \frac{1}{t}\sum_{i=1}^{t}|x(i) - \hat{x}(i)| \tag{2.2}$$

Content loss [Johnson et al., 2016] uses a pre-trained classification network (N) to measure the semantic difference between images, and how humans visualize those differences, it can be further expressed in Equation 2.3 as the Euclidean distance between the high-level representations of two images (Y and $\hat{I}$).

$$L_{cont}(I, \hat{I}, N) = \frac{1}{h_l w_l c_l} \sum_{i,j,k} (N_l^{i,j,k}(I) - N_l^{i,j,k}(\hat{I})) \tag{2.3}$$

where $N_l^{i,j,k}(I)$ represents the high-level representation extracted from the l layer of the network, h_l is the height, w_l the width and c_l is the number of channels of the feature map.

Adversarial loss in SR, follows the same objective as the content loss, but aims on improving its results, here the more realistic SR images are obtained with a GAN based network Wang et al. [2018b], Ledig et al. [2017]. As mentioned before, the generator's job is to generate SR images, and the discriminator is used to determine the authenticity of the generated samples.

The perceptual loss proposed by [Ledig et al., 2017] was formulated as the weighted sum of a content loss and an adversarial loss component (Equation 2.4).

$$L_{perceptual} = L_{cont} + 10^{-3} L_{adversarial} \tag{2.4}$$

Similarly to the content loss, this adversarial loss makes the results more realistic when human perception is taken into account. The main drawback is that these losses result in poorer performance when evaluated using the standard objective metrics (Section 2.7) [Ledig et al., 2017], leading most researchers to favor pixel loss instead.

2.7 Image Quality Assessment Methods

Image quality can have several definitions depending on the measurement methods, it is generally a measure of the quality of visual attributes and perception of the viewers. The image quality assessment (IQA) methods are categorized into subjective methods (human perception of an image is natural and of good quality) and objective methods (quantitative methods by which image quality can be numerically computed).

Subjective methods are more in line with the needs of the area, but are also time-consuming and usually expensive, thus the objective methods are currently the mainstream. However, these methods aren't necessarily consistent among themselves, as objective methods are usually unable to capture the human visual perception very accurately, which may lead to results that don't match reality. In addition, the objective IQA methods are divided into three types: full-reference methods performing assessment using

reference images [Wang et al., 2004, Horé and Ziou, 2010, Zhang et al., 2018a], reduced-reference methods based on comparisons of extracted features [Zhang et al., 2011], and no-reference methods known as blind IQA[Mittal et al., 2012], that use no reference images. However, the full-reference methods are the most used once the tendency is to compare the generated images with the original HR version.

2.7.1 Peak Signal-to-Noise Ratio

The peak signal-to-noise ratio (PSNR) [Horé and Ziou, 2010] is one of the most popular reconstruction quality measurement of lossy transformation. In images PSNR is used as a full-reference quantitative measure of the compression quality of an image.

Figure 4 shows the PSNR value of the same image on different levels of a specific degradation.

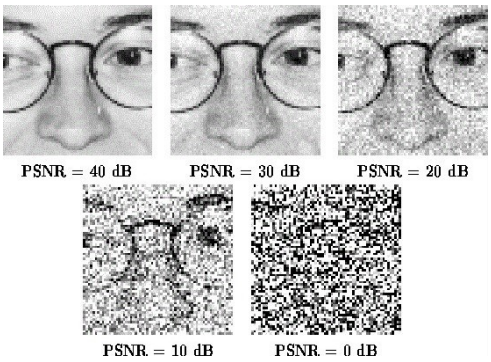


Figure 4: PSNR values with different levels of degradation [Veldhuizen]

In the field of super-resolution, the PSNR of an image is defined by the maximum pixel value and the mean square error between the HR image in comparison to the SR image. For a given maximum pixel value M (usually 255 for 8-bit color space depth), the HR image x having t pixels and the SR image $\hat{x}$, the peak signal-to-noise ratio is defined in Equation 2.5, where MSE is the mean squared error between x and $\hat{x}$, given by Equation 2.6.

$$PSNR = 10 \log_{10} \left(\frac{M^2}{MSE} \right) \tag{2.5}$$

$$MSE = \frac{1}{t} \sum_{i=1}^{t} (x(i) - \hat{x}(i))^2 \tag{2.6}$$

PSNR can be a misleading metric as it focuses on individual pixel values (Equation 2.5), without considering actual structural information like texture, edges or blur. Consequently, it is possible for an image to have a higher PSNR, indicating minimal pixel variation according with the reference image, yet still appear less accurate or visually pleasing.

Still, this metric is of today, the most used for image comparisons, especially in the field of SR where new algorithms tend to compare their results with older ones.

2.7.2 Structural Similarity Index Metric

Once PSNR does not consider the structural composition of the image, the structural similarity index metric (SSIM) was proposed [Wang et al., 2004]. This full-reference metric aims to compare the contrast, luminance, and structural details within the images.

For an image x with N pixels, the luminance L_x and contrast C_x are estimated as the mean and standard deviation of the image intensity, $L_x = \frac{1}{N}\sum_{i=1}^{N} x(i)$ and $C_x = \left(\frac{1}{N-1}\sum_{i=1}^{N}\left(x(i) - L_x\right)^2\right)^{\frac{1}{2}}$, the comparisons based on the contrast and luminance between x and $\hat{x}$, are respectively given by Equations 2.7 and 2.8, where μ_1 and μ_2 are constants used for stabilization, $\mu_1 = (K_1 L)^2, \mu_2 = (K_2 L)^2$, $K_1 = 0.01$ and $K_2 = 0.03$ and $L = 255$ for 8-bit component images.

$$Comp_L(x, \hat{x}) = \frac{2L_x L_{\hat{x}} + \mu_1}{L_x^2 + L_{\hat{x}}^2 + \mu_1} \tag{2.7}$$

$$Comp_C(x, \hat{x}) = \frac{2C_x C_{\hat{x}} + \mu_2}{C_x^2 + C_{\hat{x}}^2 + \mu_2} \tag{2.8}$$

Assuming the covariance between x and $\hat{x}$ as $\sigma_{x\hat{x}}$ and μ_3 as another stabilization constant, $\mu_3 = \mu_2/2$, the function for structural comparison is represented on Equation 2.9.

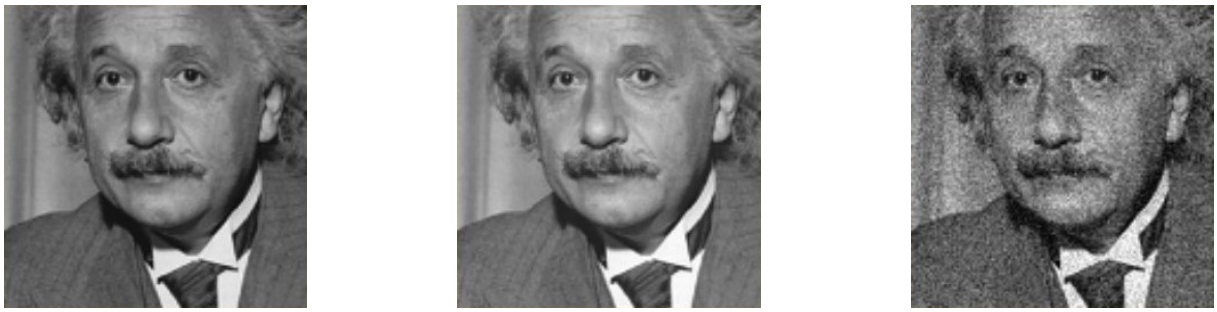
$$Comp_S(x, \hat{x}) = \frac{\sigma_{x\hat{x}} + \mu_3}{C_x C_{\hat{x}} + \mu_3} \tag{2.9}$$

Finally we can set the structural similarity index on Equation 2.10.

$$SSIM = \{Comp_L(x, \hat{x})\}^a \{Comp_C(x, \hat{x})\}^b \{Comp_S(x, \hat{x})\}^c \tag{2.10}$$

In Equation 2.10, a, b and c are control parameters that can be adjusted to change the importance of the luminance, contrast, and structural comparison in the final value, usually, $a = b = c = 1$.

SSIM was built based on the structural information within an image. As Figure 5 shows, it helps to identify those situations mentioned where PSNR can be misleading. We see that the image with a high level of noise has a better PSNR value than the one with a slight light change, different to the SSIM that rates the noisy image as worse. This metric became widely used in the SR field, in combination with PSNR for being a significant complement to result evaluation.



Original PSNR=24.32 SSIM=0.9689 PSNR=28.20 SSIM=0.6559

Figure 5: PSNR and SSIM with different levels of degradation

2.7.3 Mean Opinion Score

The Mean Opinion Score (MOS) lies in the subjective category of IQA methods, it consists of collecting human opinions or ratings. In this approach, quality assessors or testers are asked to individually grade images based on specific criteria, such as sharpness, natural appearance, color accuracy, or other relevant visual attributes and the final MOS is the mean of the rated scores, enabling its use as a full-reference, reduced-reference, or no-reference metric.

This method has limitations, such as the inconsistency between the scores; different testers might rate images differently based on their personal preferences or understanding of image quality. This can lead to variations in results, making them subjective.

In the context of SR techniques, objective IQA methods such as PSNR and SSIM might not always align with human perception, some SR methods might score well objectively but poorly in subjective evaluations. Therefore, despite using objective quality measures, researchers, like Ledig et al. [2017], also resort to MOS-based studies.

2.7.4 Other Evaluation Metrics

In addition to the IQA methods mentioned above, there are other less popular ones.

Machine learning can also be used in this context, where human behavior is learned using large datasets with distorted images or subjective scores and error maps; the downside is that these methods require more resources, especially for large datasets.

Learned Perceptual Image Patch Similarity (LPIPS) [Zhang et al., 2018a] evaluates the perceptual similarity between two images. It aims to measure how humans perceive differences between images by

using convolutional neural networks (CNNs) to capture visual features.

The Natural Image Quality Evaluator (NIQE) Mittal et al. [2012] quantifies the quality of an image solely based on its statistical properties, without needing a reference or knowledge of the original image. It works by analyzing measurable deviations or differences between the statistical regularities observed in natural images and the statistical characteristics exhibited in the image under evaluation.

The feature similarity index measure (FSIM) [Zhang et al., 2011], combines information from phase congruency and image gradient magnitude to consider specific areas of the image that the humans usually find appealing.

2.8 Models results

Table 2 lists the results of some SR models and the bicubic interpolation method (baseline) in two datasets (BSD100 and Set14) for a scale factor of 4, as well as the corresponding training datasets used.

The unsupervised model ZSSR is far from having the best results, however, a deeper comparison with the supervised approaches is required, in order to understand if it shows any advantages in different situations and conditions.

We can also see that a model outperforming other, sometimes may depend on the evaluation dataset (HAN and IPT for example), these variations can happen due to the content of the training dataset making the exploration of how does the training dataset influences the results of a SR model a valid research topic.

The best results also change according to the metric, for example, SRResNet shows the highest SSIM in BSD100 dataset, but is outperformed in PSNR by other models. Additionally, the results of a perceptual model SRGAN (based on human perception) are the lowest in the table, getting very close to the bicubic interpolation. This proximity added to the side by side comparison of SRGAN and Bicubic interpolation in Figure 6, proves that this metrics are not optimal for measuring image quality, requiring further exploration.

Model	BSD100 (PSNR/SSIM)	Set14 (PSNR/SSIM)	Training dataset	Source
SRCNN	26.68/0.7291	27.49/0.7513	ImageNet	[Dong et al., 2014]
DRCN	27.21/0.7493	28.50/0.7782	ImageNet	[Kim et al., 2016]
SRDenseNet	27.47/0.7318	28.50/0.7782	ImageNet	[Tong et al., 2017]
SRResNet	27.58/**0.7620**	28.49/0.7800	ImageNet	[Ledig et al., 2017]
SRGAN	25.16/0.6688	26.02/0.7397	ImageNet	[Ledig et al., 2017]
EDSR	27.71/0.7420	28.80/0.7876	DIV2K	[Lim et al., 2017]
ZSSR	27.12/0.7211	28.01/0.7651	———	[Shocher et al., 2018]
MSRN	27.52/0.7273	28.60/0.7751	DIV2K	[Li et al., 2018]
RCAN	27.77/0.7436	28.87/0.7889	DIV2K	[Zhang et al., 2018b]
DBPN	27.72/0.7400	28.82/0.7860	DIV2K+Flickr2K	[Haris et al., 2018]
RRDB	27.80/0.7454	28.99/**0.7917**	ImageNet	[Wang et al., 2018b]
HAN	**27.85**/0.7455	28.90/0.7890	DIV2K	[Niu et al., 2020]
IPT	27.82/——	**29.01**/——	ImageNet	[Chen et al., 2021]
Bicubic	25.94/0.6847	25.99/0.7486	———	

Table 2: SISR models for a scale factor of 4

(a) SRGAN

(b) Bicubic interpolation

Figure 6: SRGAN vs Bicubic interpolation on image from Set14 dataset

2.9 Conclusion

In recent years, the use of deep learning in image super-resolution has become the mainstream. The results achieved by deep learning models have surpassed traditional methods, offering enhanced image resolution across various applications.

Generally, as the depth and the number of parameters grow, performance improves [Yang et al., 2019]. Some researchers have directed their focus towards heavier models and excelled in achieving state-of-the-art performance, making use of a high computational demand, while others work to design lighter models that can still achieve competitive results [Gendy et al., 2023] or even the development of promising unsupervised techniques that don't rely on an external training dataset Shocher et al. [2018].

When it comes to evaluate the models results, the most commonly used metrics, PSNR and SSIM, sometimes fall short in capturing the perceptual quality of enhanced images when compared to opinion scoring, which can be impractical with large datasets. Consequently, it would be relevant to conduct a deeper study of the commonly used IQA metrics, while also exploring less well-known alternatives, such as LPIPS [Zhang et al., 2018a], to gain a broader understanding of their effectiveness.

Part II

Core of the Dissertation

Chapter 3

Method and Contribution

The work presented here aims to explore some issues related to SISR:

- Evaluating the benefits of having specificity in datasets. Three datasets were built for this purpose. One consisting of a combination of super resolution datasets commonly used for training SR models. Secondly, a Domain dataset containing only forest related images, and finally a scene specific dataset of a single forest created in Unreal Engine 5.
- Exploring loss functions, their alignment with subjective and objective metrics. For this purpose two models were trained with different loss functions (perceptual loss and pixel loss).
- Evaluating the impact of the cardinality of the training set by progressively increasing the number of training images.
- Exploring the differences of using real LR images and obtaining those through interpolated down-scaling methods on the HR images. For that, an additional pair of training and test dataset was created using original LR images, different to downsampled LR images from the previous datasets.
- Comparing supervised and unsupervised methods both regarding output quality and other factors, such as the training effort.
- Exploring the problem of the evaluation effort among SR results. We used the standard image quality assessment metrics, as well as human perception and a less popular metric, to understand at what point do this different methods align with each other.

3.1 Datasets

To study the impact of specificity of the datasets, training was performed on three distinct datasets with different levels of specificity: a general dataset, a domain specific dataset with real images, and an even

more specific synthetic dataset created in Unreal Engine.

Furthermore, to explore the impact of the cardinality of the datasets, each version had 5 different cardinalities, totaling 15 different training datasets.

Based on the DIV2K dataset, which contains 800 images, this was set as the smallest cardinality. The remaining cardinalities were obtained doubling iteratively the number of samples, resulting in cardinalities: 800, 1600, 3200, 6400, and 12800 images.

For the general domain dataset, in the smallest cardinality, 800 images, it is identical to DIV2K. Images from other datasets (Flick2K, OutdoorScene), as well as images retrieved from Flickr, were added to fulfill the larger cardinality.

For the specific domain the forest theme was chosen mainly for its high level of detail, as well as for the abundance of images. Images for these datasets were retrieved from Flickr and subsequently filtered the ones considered blurred to ensure the dataset's quality.

For the most specific dataset a synthetic forest environment was used in Unreal Engine 5 (Electric Dreams Environment Games [2023]). Images were captured from random camera positions above the ground, and were subsequently reviewed to eliminate any possible major occlusions with objects.

To facilitate the distinction between the datasets, these will be referred henceforth as "General", "Domain", and "Unreal" in increasing order of specificity. It is also important to mention that for these three datasets, the LR images were obtained by downscaling the HR images using bicubic interpolation.

In real life, it is hard and resourceful to create an image dataset with HR images followed by the corresponding true LR images. The common procedure is to use downscaling methods on the HR images to obtain the LR correspondence [Dong et al., 2014, Shocher et al., 2018, Ledig et al., 2017, Niu et al., 2020]. This means that the models are trained to perform the inverse process of the downscaling methods instead of enhancing true LR images. This raises the question of whether, in a real-world application, we would achieve better results using true HR-LR image pairs.

To explore this issue, we created the "True LR" training dataset with the same forest [Games, 2023] in Unreal Engine 5 but changing its resolution between screenshots to obtain the LR images. These simulated true LR images allow us to understand the impact of using downsampled vs true LR. The True LR dataset also includes a test set of 10 images obtained with the same process.

All the datasets used are listed in Table 3 and can also be found here:

https://github.com/lucasCarvalho64/Datasets-for-Super-Resolution.git

Table 3: Datasets used/created

Type	Name	Cardinality	Specificity	Downscale Process
Train set	General	800...12800	Multiple domains	Bicubic
	Domain	800...12800	Forest scenarios	Bicubic
	Unreal	800...12800	Specific forest	Bicubic
	True LR	12800	Specific forest	Graphics Engine
Test set	Set14	14	Multiple domains	Bicubic
	Forest	10	Forest scenarios	Bicubic
	Unreal	10	Specific forest	Bicubic
	True LR	10	Specific forest	Graphics Engine

3.2 Supervised Models

SR models have improved over time as new strategies and deeper networks are proposed. These improvements are usually accompanied by higher computational demands and increased training times. The purpose here is not to evaluate the models themselves, hence, the models used in here were selected mainly for representing different facets of the SISR current approaches.

Super Resolution GAN (SRGAN) Ledig et al. [2017], proposed a generative adversarial network with the objective of creating more perceptually realistic images, in which the generator was referred to as Super Resolution Residual Network (SRResNet). In SRGAN, the SRResNet is trained with the goal of fooling a discriminator that learns to distinguish super-resolved images from true HR images. The generator is composed of 16 residual blocks. The residual block has become the basic unit in these network structure consisting of two 3×3 convolutional layers, two batch normalization layers, and a ReLU activation function in between.

For the training process, in Ledig et al. [2017] the authors first train only the SRResNet with MSE (Section 2.7) as the loss function and employ it as initialization for the SRGAN's generator, this way the discriminator receives proper SR images from start. The SRGAN loss function was formulated as the weighted sum of a content loss and an adversarial loss component (Section 2.6). Here, we followed the same training strategy, resulting in a pixel-based model (SRResNet) and a perceptual-based model (SRGAN).

For more detailed information about these networks, we recommend reading the original paper Ledig et al. [2017], as our work follows the implementation described therein.

3.3 Unsupervised Approach

Supervised SR networks trained on large and diverse collections of LR and HR image examples aim to capture the vast variety of all possible LR-HR relationships. Consequently, these networks tend to be extremely deep and complex. In contrast, the diversity of LR-HR relationships within a single image is much lower, allowing for encoding with a much smaller architecture in a simpler image-specific model. This is the idea behind ZSSR Shocher et al. [2018].

The network has 8 hidden layers, each containing 64 channels and ReLU activations. Similar to the supervised approach, it only learns the residual between the interpolated LR and its HR correspondence.

While the methods in the previous section required a training dataset, ZSSR takes as its only input the LR image to be upsampled. Based on that image, the creation of a small training dataset is necessary. Data augmentation is used to generate LR-HR pairs through multiple downscales of the input. The dataset is further expanded by applying four 90° rotations along with vertical and horizontal mirror reflections, significantly increasing the number of unique examples derived from the same image.

Lastly, it uses a method similar to a self-ensemble: it generates 8 different outputs for the 8 rotations and flips of the test image and combines them. It is further combined with the back-projection technique, so that each of the 8 output images undergoes several iterations of back-projection and finally the median image is corrected by back-projection as well.

L1 loss (Equation 2.2) is used with ADAM optimizer. Learning rate starts on 0.001 and training stops when it reaches 10^{-6}. A linear fit of the reconstruction error is periodically taken. If the standard deviation is greater by a factor of 10 than the slope of the linear fit we divide the learning rate by 10.

As detailed in Shocher et al. [2018], using a gradual increase in resolution has an impact on the model's performance, hence, in Section 4.2 we compare both approaches.

Chapter 4

Outcome Assessment and Discussion

In this chapter, we provide a comprehensive analysis of the results obtained from all models introduced in the previous chapter. The performance of each model is evaluated through both quantitative metrics and qualitative visual assessments across different datasets and conditions.

All images were evaluated using the standard metrics for SR, PSNR and SSIM, in combination with the LPIPS metric (Section 2.7).

While PSNR calculates the ratio between the maximum possible pixel value and the difference (error) between the original and reconstructed image, SSIM assesses image quality by focusing on structural information, contrast, and luminance changes between the images. Higher PSNR and SSIM values indicate better quality.

Different from the first two, LPIPS computes the distance between the feature representations of image patches by pre-trained CNNs. Meaning that a higher LPIPS score indicates a bigger difference between images. All metric values presented are a mean of 5 different runs on the same conditions.

Furthermore, each visual example was also evaluated based on human perception. A user study was conducted with 40 participants, who were asked to assess each image. Participants were tasked with selecting the sample they believed was the most similar to the original image or indicating if the images were too similar to make a clear choice.

4.1 Supervised Methods

For each domain dataset (General, Domain and Unreal), we tested 5 size variations, starting with 800 images and doubling the number each test, up to 12800 images, on SRResNet and SRGAN. All networks were trained on a NVIDIA RTX 3060 with a learning rate of 10^{-4} and a batch size of 16, using 192x192 HR images, always for a scale factor of 4. The LR images were generated by downsampling the HR images using bicubic interpolation, except for the True LR dataset (generated by the graphics engine).

The SRResNet networks were trained for 10^6 update iterations and served as an initialization for the SRGAN networks, which were further trained for 10^5 update iterations.

The three levels of specificity were evaluated on two distinct test sets. The first dataset consists of a sample on 10 images from the Unreal forest dataset, different from the images used during training (Unreal test set). The second is the Set14 Zeyde et al. [2012], a general benchmark test dataset with 14 images widely-used for SR tasks. Furthermore, the Unreal and True LR models were compared using the True LR test set, which is also composed by 10 images. Finally, in order to further explore the unsupervised approach, a 10 image test set of the Domain dataset was also used.

4.1.1 General Dataset

In this section we present the results of the evaluation of the models trained with General dataset, see Table 4. Regarding the metrics, results show a relative agreement between all the metrics when considering the Set14 test set. Results tend to get better as the number of samples in the dataset increases, although there are no significant improvements.

Considering the Unreal test set, the differences as the dataset increases are more significant for the SRGAN. However, for the Unreal test set we can observe a discrepancy between the metrics as the dataset increases, with LPIPS tending to get worse results, while both SSIM and PSNR results keep improving. Nevertheless, the only significant improvement that can be observed in this test set is for the model trained with the SRGAN, with the SSIM metric going from 0.6508 to 0.6810.

Since we are testing in a specific domain, and taking into account that the General datasets do not have a significant number of forest images, it seems reasonable to expect an increase in performance in some metrics as the number of samples in the dataset goes up.

Another interesting take is that even when considering the smaller datasets, the SRResNet model achieves a higher SSIM and PSNR scores than SRGAN. This is to be expected as SRResNet uses pixel based loss, whereas SRGAN uses content based loss. These results are in line with the results presented in Ledig et al. [2017] where SRResNet also obtains a higher SSIM and PSNR score than SRGAN. Worth noting is that in Ledig et al. [2017] the authors also use Mean Opinion Score (MOS) as an evaluation metric. The reported results indicate that MOS is not aligned with SSIM or PSNR, with SRGAN having a higher MOS score. Our results show the same tendency, with SRGAN having a better LPIPS score than SRResNet, hinting at LPIPS higher affinity with human perception. This can also be seen in Figure 7 that shows side by side an image from Set14 upsampled by SRGAN and SRResNet, both with 12800 training images, followed by Figure 8 which shows the respective study results. Its clearly noticeable that SRGAN

produces the most detailed images to the human eye, having 82.5% of the votes.

Figure 9 shows an evaluation HR sample with closeup regions that can shed some light on the misalignment between the metrics. The model trained with the highest cardinality dataset provides results that look blurrier than when training with the smallest dataset. On the other hand, in particular when looking at the region with branches, the sharper image also has clearly visible artifacts. SSIM and PSNR do rate the 12800 dataset images higher, whereas LPIPS considers them worse. This example clearly shows that different metrics evaluate different features. Figure 10 shows that 85% of the users rated the sample from the 800 images training model has as the best image, reaffirming LPIPS as a better perceptual metric.

Table 4: Results from the models trained with the **General** dataset

Model	**Test set**	**Images**	**SSIM**	**PSNR**	**LPIPS**
SRResNet	**Set14**	800	0.7910 ±0.0011	27.96 ±0.03	0.2125 ±0.0017
		1600	0.7960 ±0.0007	28.15 ±0.05	0.2134 ±0.0012
		3200	0.7992 ±0.0006	28.34 ±0.01	0.2119 ±0.0006
		6400	0.8022 ±0.0003	28.49 ±0.01	0.2103 ±0.0005
		12800	**0.8042 ±0.0003**	**28.62 ±0.03**	**0.2097 ±0.0008**
	Unreal	800	0.6915 ±0.0005	26.62 ±0.011	**0.2286 ±0.0012**
		1600	0.6900 ±0.0006	26.76 ±0.01	0.2332 ±0.0002
		3200	0.6946 ±0.0092	26.81 ±0.07	0.2302 ±0.0056
		6400	0.6992 ±0.0002	26.94 ±0.01	0.2350 ±0.0001
		12800	**0.7006 ±0.0007**	**26.99 ±0.02**	0.2346 ±0.0010
SRGAN	**Set14**	800	0.7393 ±0.0105	26.56 ±0.18	0.1730 ±0.0043
		1600	0.7420 ±0.0049	26.61 ±0.19	0.1716 ±0.0046
		3200	0.7564 ±0.0107	27.05 ±0.40	0.1725 ±0.0109
		6400	**0.7612 ±0.0068**	27.47 ±0.13	0.1676 ±0.0050
		12800	0.7591 ±0.0082	**27.71 ±0.26**	**0.1641 ±0.0072**
	Unreal	800	0.6508 ±0.0056	25.996 ±0.12	**0.2135 ±0.0028**
		1600	0.6536 ±0.0120	26.203 ±0.30	0.2150 ±0.0045
		3200	0.6736 ±0.0029	26.48 ±0.12	0.2224 ±0.0037
		6400	0.6723 ±0.0122	26.54 ±0.24	0.2282 ±0.0057
		12800	**0.6810 ±0.0052**	**26.74 ±0.13**	0.2239 ±0.0054

SRGAN SRResNet

Figure 7: SRResNet vs SRGAN with 12800 training images on a Set14 image.

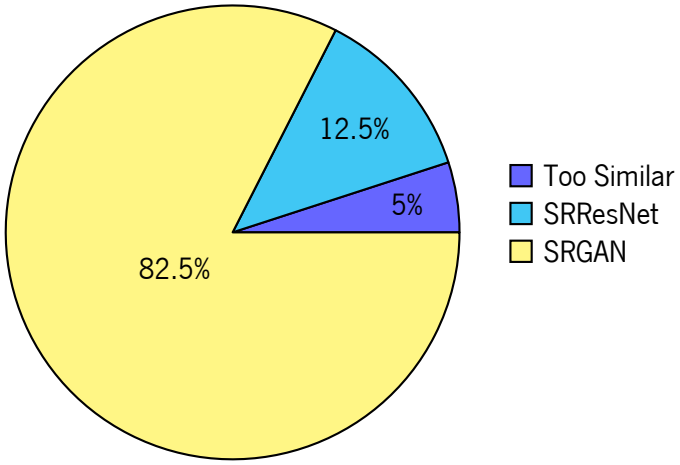


Figure 8: Perceptual Results from Figure 7.

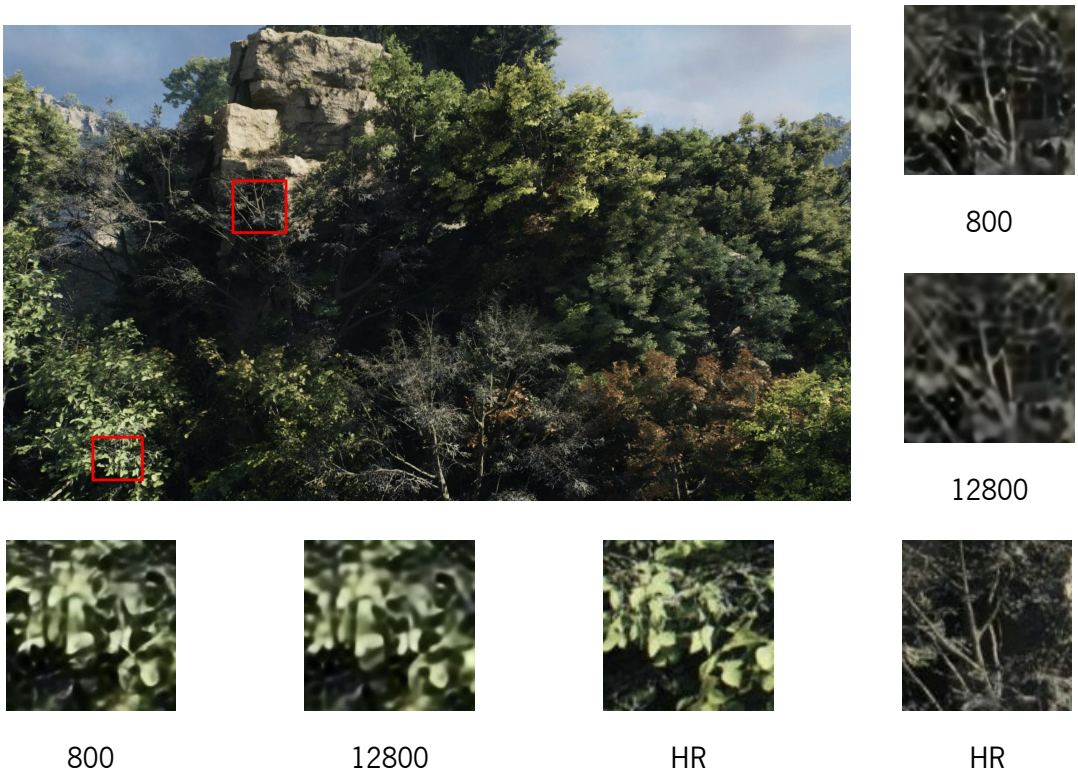


Figure 9: General SRResNet results with 800 and 12800 training images on a Unreal test set image.

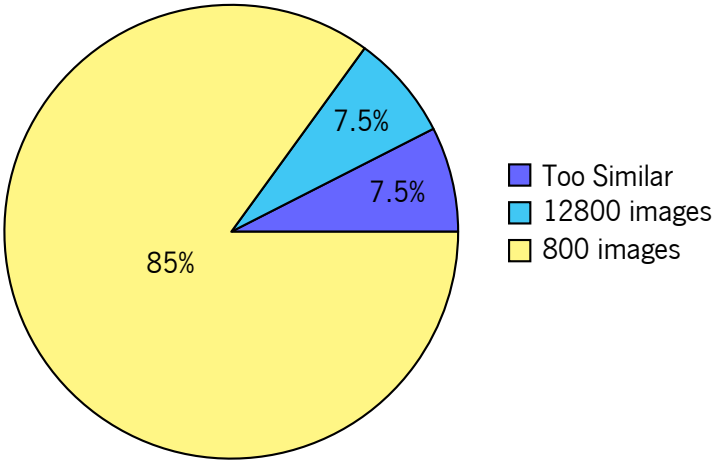


Figure 10: Perceptual Results for Figure 9.

4.1.2 Domain Dataset

The results of the models trained on the Domain dataset are presented in Table 5. As expected, due to the specificity of the training dataset, the results on Set14 suffer significantly when compared with models trained with the General dataset.

For the Unreal dataset it can be observed that the SRGAN's results for SSIM and PSNR suffer a significant reduction after 800 images, apparently overfitting the training data. Note that SRGAN includes the SRResNet, and further trains it. This extended training, along side having a niched dataset instead of a general one could lead to overfitting. However, the LPIPS metric says otherwise. Instead of having the same variations as PSNR and SSIM, it progressively improves as we add more training data.

Considering SRResNet, tests performed on the Unreal test set, when comparing with the models trained with the General dataset, show an improvement in both PSNR and SSIM. The 800 training images model already exceeds the previous domain with 12800 images. On the other hand, the LPIPS metric has no significant change. Regarding the size of the training set, SRResNet consistently improves PSNR and SSIM with higher cardinalities, while LPIPS gets worse.

Figure 11 illustrates the progressive changes in the same HR image produced by SRResNet as the size of the training dataset increases. We notice that the results tend to become slightly blurred, but the smaller branches look more defined from 1600 forward. These contrasting changes do not allow a definitive statement on what image looks better. This is also the majority opinion of the human evaluators, as can be seen in Figure 12. The differences are rated as very subtle, making most users rate the images as too similar. The second and third most voted options, were the 12800 and the 800 images respectively. This highlights once again, that pixel based metrics and perceptual metrics analyze different important aspects of the images. Considering that LPIPS tends to penalize blur, the visual inspection is in line with the scores obtained for this metric.

In Figure 13 we can see how the result varies with the training set size for SRGAN. From 800 to 3200, image sharpness increases, but more artifacts are also present in 3200 dataset. At 6400 images, the results seem blurrier, while getting sharper again with less artifacts with the 12800 images. Besides these variations, SSIM scores agree with this last statement, while PSNR still rates the model trained with the 800 dataset with a slightly higher score. Considering the perceptual results from Figure 14, the most voted images were the 3200 and 12800. hinting that humans not only penalize blur but also visual artifacts.

Additionally, Figure 15 compares a full image from the Unreal test set, generated from both SRResNet and SRGAN, along side the true HR image. As the perceptual results shown in Figure 16 present, it follows the same tendency as before, with SRGAN having the most votes. On one hand, SRGAN is able to capture more detail and sharpness, presenting us a more visual appealing image, however, when comparing with the real image, SRGAN also tends to deviate more from the original, in some places of small vegetation, the model fills the lost details with random branches that could make sense in the image. On the other hand, SRResNet does not fill the image with information that its not there, it keeps its loyalty to the original

sample at the cost of a blurrier image with lower levels of detail.

Table 5: Models trained with the **Domain** Dataset

Model	Test set	Images	SSIM	PSNR	LPIPS
SRResNet	**Set14**	800	0.7393 ±0.0044	26.89 ±0.04	**0.2786 ±0.0028**
		1600	0.7459 ±0.0004	27.09 ±0.01	0.2852 ±0.0021
		3200	0.7479 ±0.0007	27.18 ±0.03	0.2906 ±0.0006
		6400	0.7514 ±0.0005	27.37 ±0.01	0.2957 ±0.0003
		12800	**0.7534 ±0.0006**	**27.49 ±0.01**	0.2964 ±0.0019
	Unreal	800	0.7169 ±0.0027	27.27 ±0.01	**0.2297 ±0.0033**
		1600	0.7221 ±0.0002	27.39 ±0.01	0.2333 ±0.0009
		3200	0.7249 ±0.0001	27.50 ±0.01	0.2386 ±0.0006
		6400	0.7278 ±0.0004	27.59 ±0.01	0.2417 ±0.0019
		12800	**0.7350 ±0.0103**	**27.63 ±0.06**	0.2553 ±0.0039
SRGAN	**Set14**	800	**0.7133 ±0.01235**	**26.28 ±0.25**	0.2842 ±0.0118
		1600	0.6893 ±0.0180	26.03 ±0.28	0.2580 ±0.0102
		3200	0.6614 ±0.0011	25.27 ±0.04	0.2548 ±0.0115
		6400	0.6811 ±0.0164	25.95 ±0.41	**0.2410 ±0.0169**
		12800	0.6825 ±0.0325	25.59 ±0.35	0.2493 ±0.0208
	Unreal	800	0.6560 ±0.0296	**26.26 ±0.40**	0.2380 ±0.0126
		1600	0.6330 ±0.0323	25.61 ±0.32	0.2260 ±0.0158
		3200	0.6152 ±0.0116	25.67 ±0.27	0.2243 ±0.0213
		6400	0.6473 ±0.0268	26.18 ±0.41	0.2181 ±0.0148
		12800	**0.6764 ±0.0229**	26.22 ±0.31	**0.2046 ±0.0093**

Figure 11: Domain SRResNet from 800 to 12800 training images on Unreal test set image.

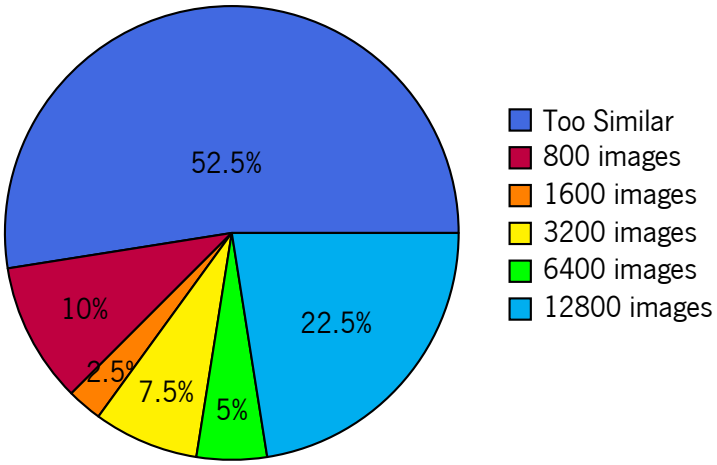


Figure 12: Perceptual Results for Figure 11.

Figure 13: Domain SRGAN from 800 to 12800 training images tested on Unreal test set image.

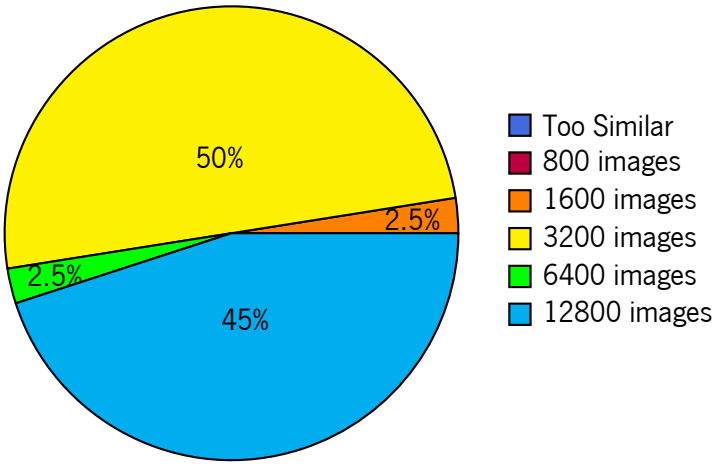


Figure 14: Perceptual Results for Figure 13.

Original

SRResNet

SRGAN

Figure 15: Domain SRResNet vs SRGAN with 12800 training images on image from Unreal testset.

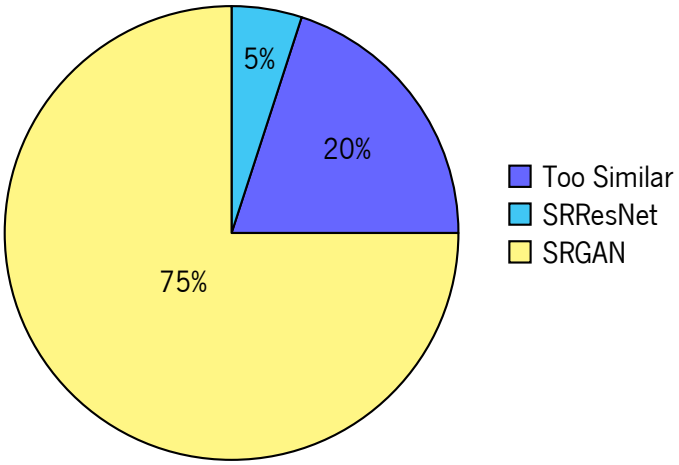


Figure 16: Perceptual Results for Figure 15.

4.1.3 Unreal dataset

For the last level of specificity, Table 6 lists the results of all models trained with the Unreal dataset. Overall, the metrics behavior for Set14 don't differ significantly from the results obtained with the Domain dataset.

The size of the dataset also follows the same pattern as the previous models, even SRGAN keeps showing significant variations for PSNR and SSIM.

Considering the Unreal test set there is a slight improvement in the pixel based metrics when testing with SRResNet. However, the SRGAN model's best results for the Unreal test set are lower than those obtained with models trained with the General and Domain training sets. Still, the best LPIPS score for this test dataset is obtained with this training data.

Looking at HR samples generated with these models can provide further insight into the results. Figure 17 and Figure 19 show samples generated with each of the models. In Figure 17 it can be observed that the upsampled images obtained with SRResNet models trained on Unreal and Domain datasets are practically identical, further hinted by the subjective results from Figure 18, where 80% of the users classified the images as too similar, which aligns considerably with the numeric data as there were no significant improvements. Figure 19, showcasing upsampling using the SRGAN, tells a different story. On one hand, for the 12800 case, all metrics present extremely close results considering the Unreal vs the Domain training set. However, the figure clearly shows severe artifacts when considering the Unreal training set. Furthermore, taking in account the results from Figure 20, the Domain training set has 90% of the votes.

This example clearly raises the issue of the suitability of these metrics to independently evaluate quality

of a SR model.

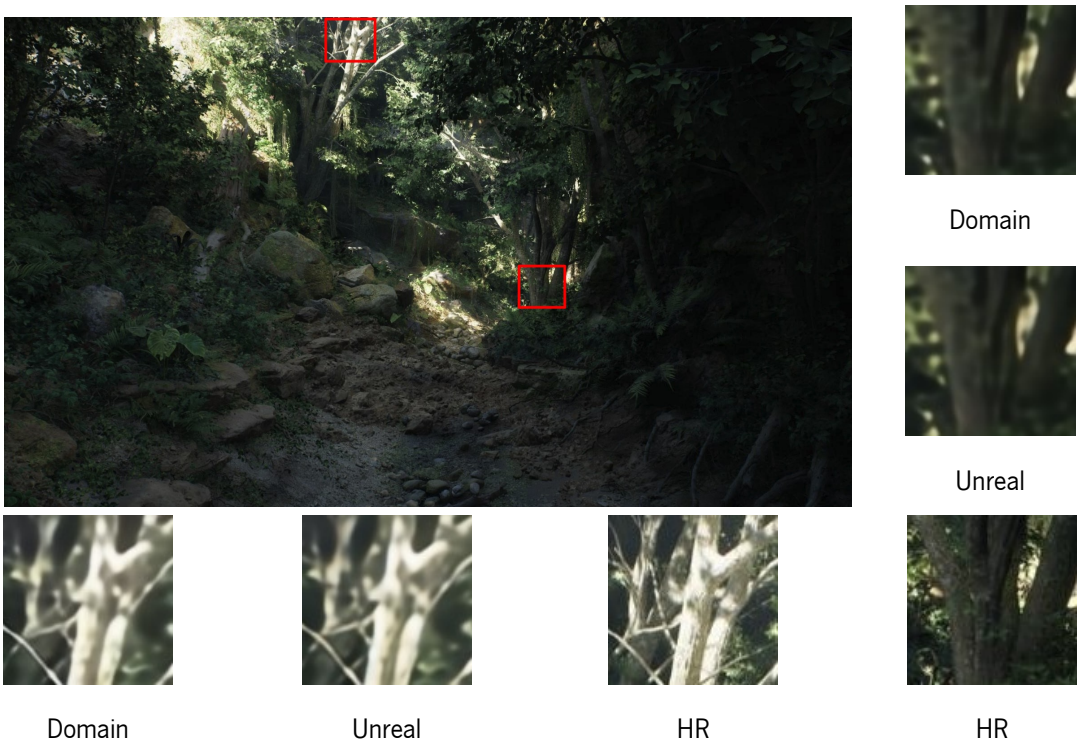


Figure 17: Domain vs Unreal SRResNet results with 12800 training images on the same image from the Unreal test dataset.

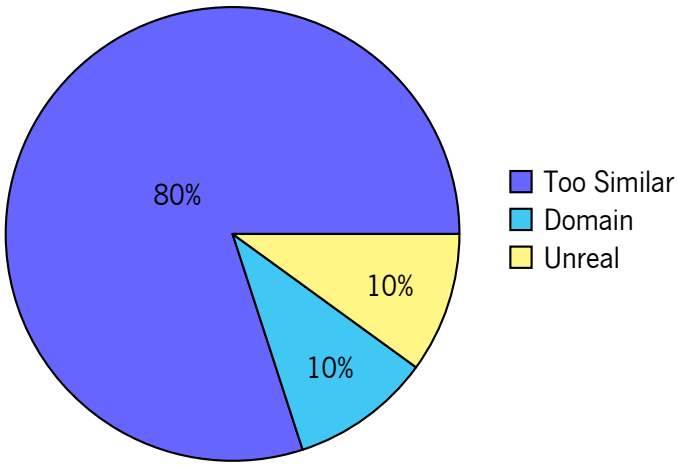


Figure 18: Perceptual Results for Figure 17.

Table 6: Models trained with the **Unreal** Dataset

Model	Test set	Images	SSIM	PSNR	LPIPS
SRResNet	**Set14**	800	0.7084 ±0.0005	26.24 ±0.07	0.3045 ±0.0034
		1600	0.7152 ±0.0016	26.46 ±0.01	**0.3017 ±0.0015**
		3200	0.7307 ±0.0006	26.74 ±0.08	0.3081 ±0.0008
		6400	0.7348 ±0.0025	26.88 ±0.04	0.3111 ±0.0013
		12800	**0.7426 ±0.0021**	**27.07 ±0.07**	0.3130 ±0.0018
	Unreal	800	0.7157 ±0.0001	27.17 ±0.01	**0.2232 ±0.0013**
		1600	0.7220 ±0.0004	27.36 ±0.02	0.2304 ±0.0017
		3200	0.7282 ±0.0002	27.51 ±0.01	0.2361 ±0.0008
		6400	0.7324 ±0.0001	27.65 ±0.00	0.2389 ±0.0006
		12800	**0.7350 ±0.0001**	**27.74 ±0.00**	0.2409 ±0.0001
SRGAN	**Set14**	800	0.6801 ±0.0146	25.15 ±0.52	0.3397 ±0.0121
		1600	0.6773 ±0.0238	24.56 ±0.70	0.3391 ±0.0151
		3200	0.6790 ±0.0160	24.88 ±0.36	0.3341 ±0.0090
		6400	0.6893 ±0.0070	25.57 ±0.08	**0.2852 ±0.0113**
		12800	**0.7036 ±0.0036**	**25.83 ±0.17**	0.2895 ±0.0093
	Unreal	800	0.6597 ±0.0153	26.08 ±0.31	0.2294 ±0.0087
		1600	0.6691 ±0.0435	26.12 ±0.43	0.2308 ±0.0059
		3200	0.6415 ±0.0073	25.76 ±0.11	0.2169 ±0.0043
		6400	0.6588 ±0.0043	26.19 ±0.05	0.2084 ±0.0030
		12800	**0.6744 ±0.0120**	**26.20 ±0.28**	**0.2007 ±0.0044**

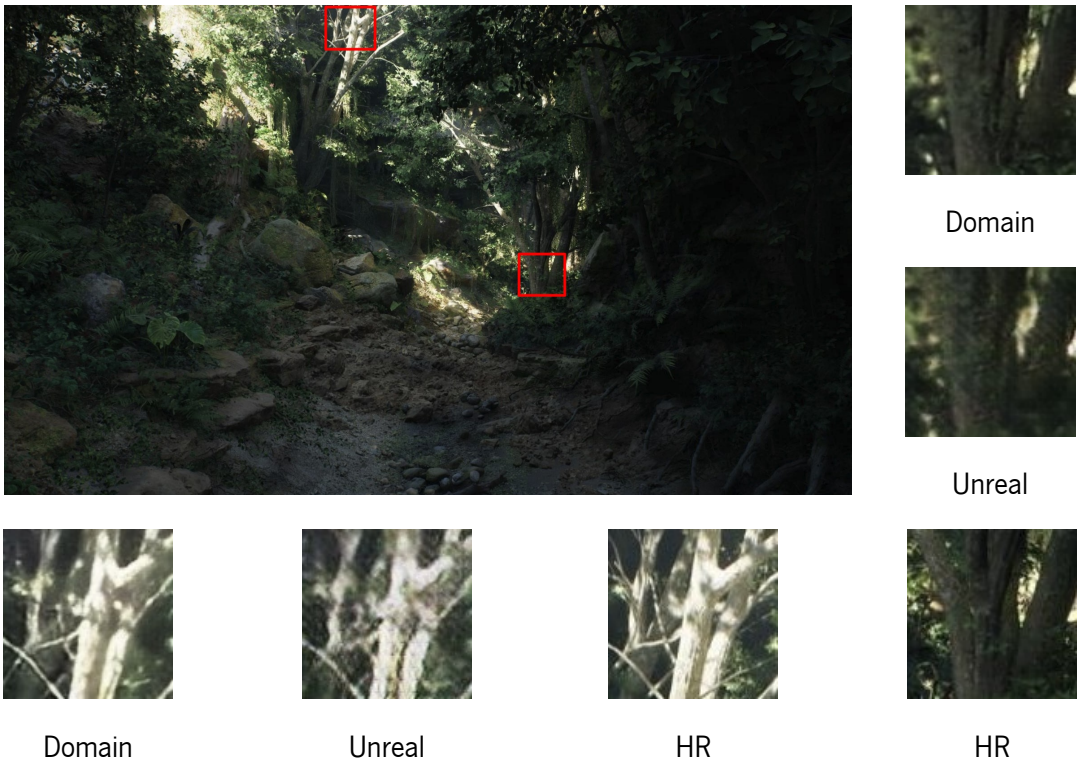


Figure 19: Domain vs Unreal SRGAN results with 12800 training images from the same image from the Unreal test dataset.

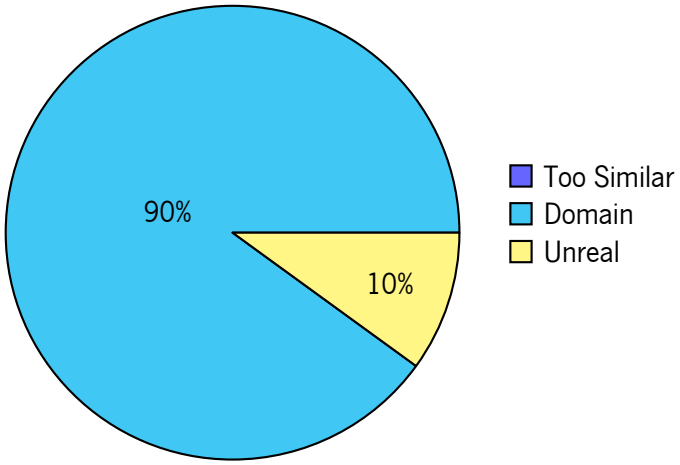


Figure 20: Perceptual Results for Figure 19.

4.1.4 True LR Dataset

In this section we compare the results of the networks trained with the True LR and Unreal datasets with 12800 images, tested on the True LR test set, see Table 7. Note that the only difference between the two training sets is the way that the LR samples were obtained. As mentioned before, we used the Unreal Engine 5 to create true LR images, hence true LR-HR image pairs of the same synthetic forest.

There are two main takeaways from the results we obtained.

First, there is a considerable decrease in performance when the input is a true LR image vs a downsampled version of a HR image (Table 6 best results for PSNR/SSIM/LPIPS: 0.7350/27.74/0.2007). This implies that real results, when one is actually uploading a true LR image, may fall short of the benchmark results published in papers. Still, it is worth noting that the comparison among the methods themselves remains valid regardless of the test set. Hence, this comparative analysis does not contradict the respective analysis of papers in this area.

The performance decrease when using training sets with true LR images is definitely worth researching. Is this particular to images generated artificially, i.e., would we get the same results in real scenarios?

Another possibility is related to the way the LR images are commonly generated. The bicubic interpolation method smooths image textures and edges. In this test set, instead of applying a filter to reduce resolution, LR images are generated by the graphics engine. True LR images are sharper and this can make the images more challenging for these models.

Secondly, although trained on true LR images, in almost all of the metrics these models performed worse than the models trained with downsampled HR images even though the input is a true LR image. Is this a feature of these models, in the sense that the models actually learn better with downsampled images? Again further research is required to clarify these issues.

Figure 21 shows a zoomed in visual example of all models tested on the same image from the True LR test set, which aligns with the numeric data (except PSNR) and the perceptual results from Figure 22, where 77.5% of the users voted for Unreal SRResNet. For the case of the SRGAN, the artifacts and noise are still present, however, the True LR training set introduced even more. Figure 23 alongside its perceptual data from Figure 24 further illustrates this point even without zoom. SRResNet trained on the True LR dataset appears blurry in certain areas, while the model trained with downsampled LR images successfully captures those details, securing 95% of the overall votes.

Table 7: True LR vs Unreal dataset models, tested on the **True LR** test set

Model	**Training set**	**SSIM**	**PSNR**	**LPIPS**
SRResNet	True LR	0.6556 ±0.0012	**22.01 ±0.07**	0.3182 ±0.0006
	Unreal	**0.6578 ±0.0009**	21.67 ±0.03	**0.3044 ±0.0018**
SRGAN	True LR	0.5935 ±0.0313	21.06 ±0.29	0.2864 ±0.0132
	Unreal	**0.6064 ±0.0102**	**21.40 ±0.23**	**0.2759 ±0.0072**

Figure 21: True LR vs Unreal dataset models, tested on image from the True LR training set.

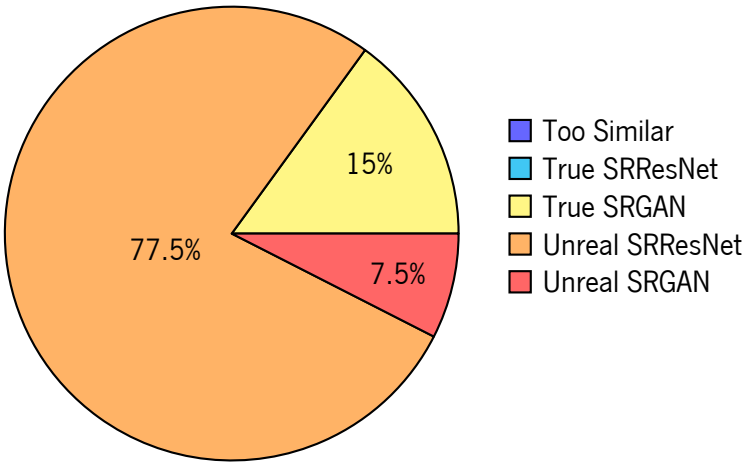


Figure 22: Perceptual Results for Figure 21.

Unreal SRResNet

True SRResNet

Figure 23: Unreal vs True LR SRResNet on image from True LR test set.

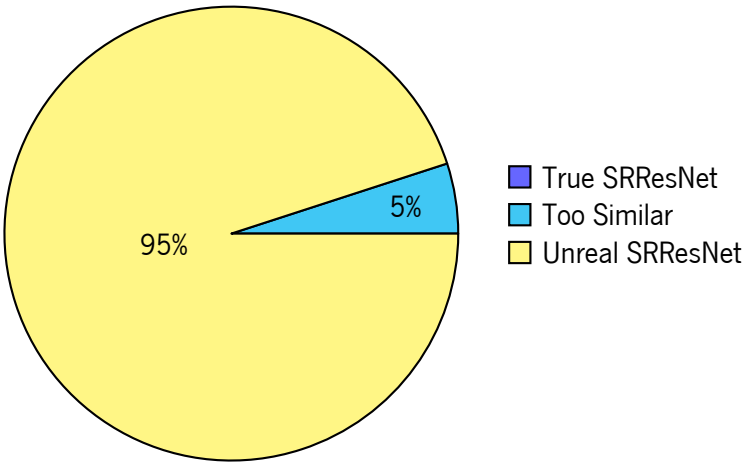


Figure 24: Perceptual Results for Figure 23.

4.2 Results of the Unsupervised Approach

This section presents in Table 8 the results of the two ZSSR variants for a scale factor of 4. Based on these quantitative metrics, although there is an overall improvement using the progressive approach, it is almost negligible. However, a visual inspection of an upsample performed with the two versions of ZSSR, see Figure 25, clearly shows that the results of the progressive approach have less artifacts, and better quality. As further demonstrated by the subjective results in Figure 26, 72.5% of users selected the progressive approach as the best. However, a gradual increase using 5 intermediate scale-factors increased the runtime from 1 minute to 6 minutes per image making this approach unfeasible to process large quantities of images. While SRResNet takes about 12 hours to train, and another 10 hours for SRGAN, but once trained it takes close to one second to generate one SR image. Furthermore, when testing with the synthetic images, all models took about twice the normal time per image including ZSSR.

When we compare the quantitative results for Set14 against the supervised models, we can observe that it is only clearly surpassed by SRResNet trained with the General dataset. Figure 27 shows the upsampling result for all models in a Set14 image. Although it is impressive that ZSSR achieves a good result based on a single image, SRResNet gets slightly more defined results. SRGAN produces the crispiest result, although the pigmentation in the petals is clearly off target. However, as shown in Figure 28, this lack of pigmentation did not affect user preference, as 97.5% of participants still rated the SRGAN model as producing the best image.

On the True LR test set, the results are way more competitive, even surpassing the supervised models on the LPIPS metric. The visual example provided by Figure 29, aligns with the LPIPS scores. In Figure

30 we see that 90% of the users took preference over the image produced by ZSSR. The unsupervised approach is able to capture some details that SRResNet cant, but a small level of aliasing is also present in the images. A contrasting example is shown in Figure 31, where aliasing became a more significant issue. In this second comparison, 82.5% of users rated SRResNet image as the best.

It is clear that ZSSR's results are closer to the supervised methods when tested with true LR images. According to Shocher et al. [2018], their method "leverages on the power of the cross-scale internal recurrence of image-specific information". Apparently, true LR images share more of this property than the downscaled ones.

Apart from the True LR dataset, a result that stands out is the score difference for Set14 and Unreal test sets. In order to further explore this issue a third test set was created, this time with 10 images from the Domain dataset. The results for this test set, presented in Table 9, show a difference to the other two test sets, sitting in the middle for both PSNR and SSIM. Results for LPIPS are closer to Set14 than to Unreal. The fact that ZSSR performs better with images from real forests than Unreal generated ones, might be significant. Can it be the case that forest generated in Unreal share less of the property that ZSSR takes advantage? Are synthetic forests too random? These are interesting research questions regarding the procedural modeling of forest like scenarios, but lying outside the scope of this paper.

Table 8: Results from **ZSSR**

Test set	**Progressive**	**SSIM**	**PSNR**	**LPIPS**
Set14	No	0.7712 ±0.0004	27.26 ±0.02	**0.2386 ±0.0007**
	Yes	**0.7791 ±0.0005**	**27.40 ±0.02**	0.2391 ±0.0004
Unreal	No	0.6861 ±0.0002	26.19 ±0.02	0.2613 ±0.0019
	Yes	**0.6886 ±0.0003**	**26.48 ±0.02**	**0.2589 ±0.0015**
Real	No	0.6446 ±0.0005	21.35 ±0.01	0.2563 ±0.0018
	Yes	**0.6459 ±0.0008**	**21.48 ±0.02**	**0.2556 ±0.0026**

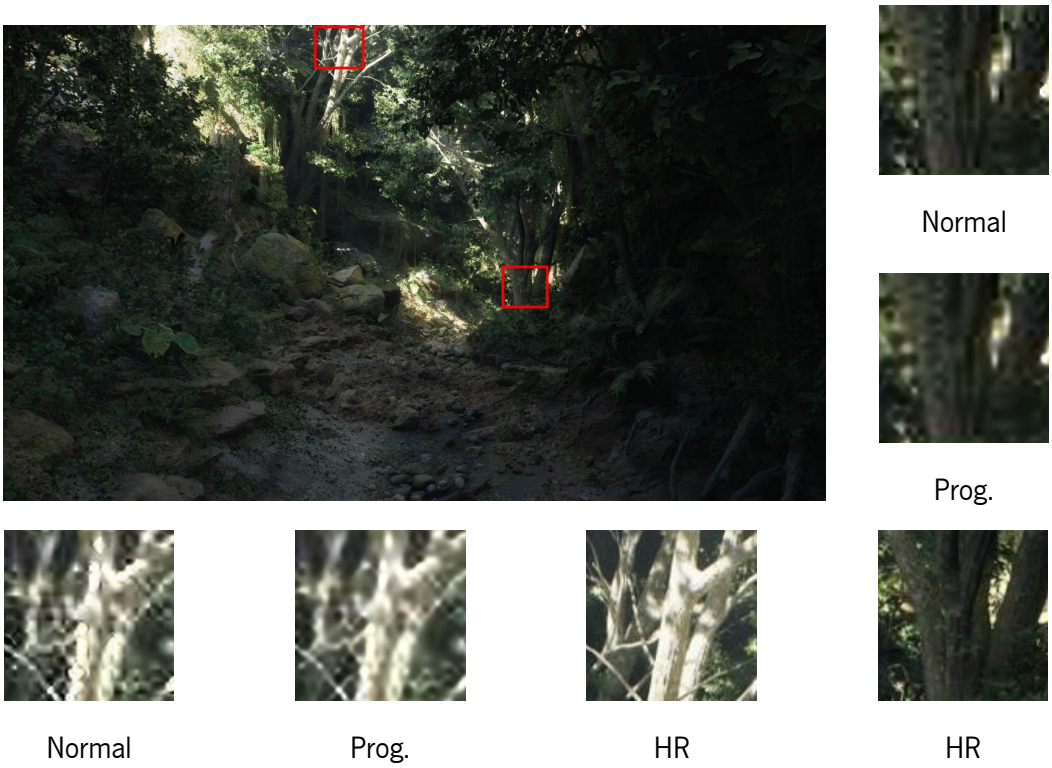


Figure 25: Normal vs Progressive ZSSR results from the same image of the unreal test set.

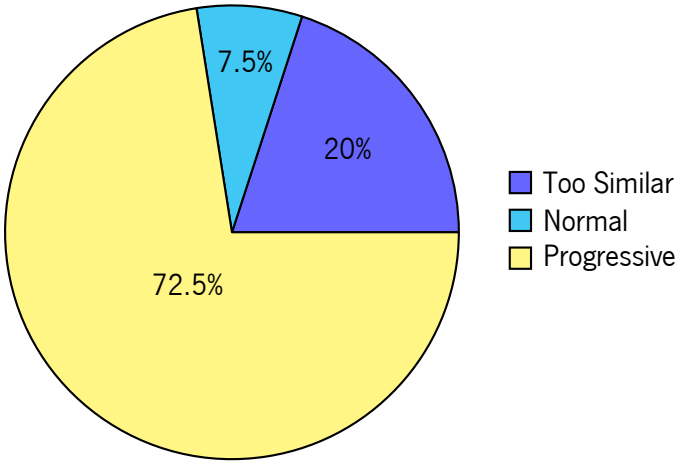


Figure 26: Perceptual Results for Figure 25.

Figure 27: ZSSR vs General SRResNet vs SRGAN results on Set14 image.

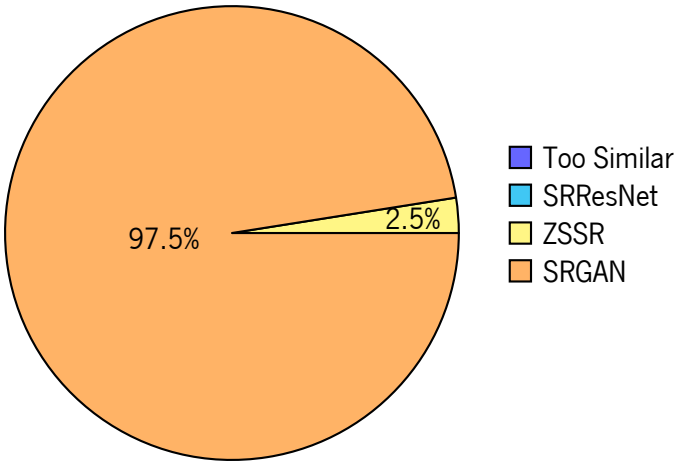


Figure 28: Perceptual Results for Figure 27.

Figure 29: Unreal SRResNet vs ZSSR, tested on image from the True LR test set.

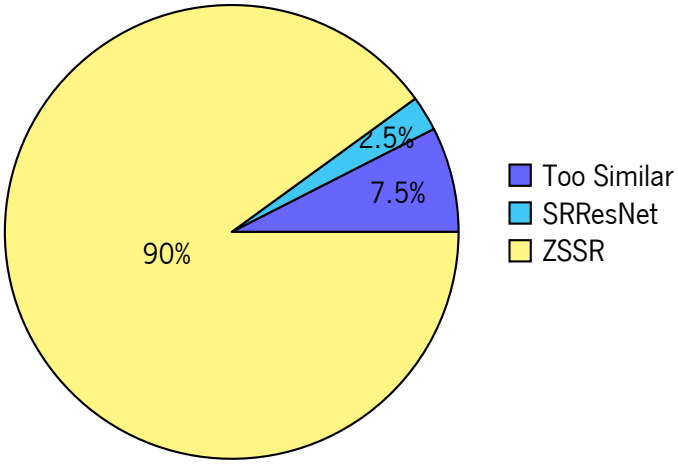


Figure 30: Perceptual Results for Figure 29.

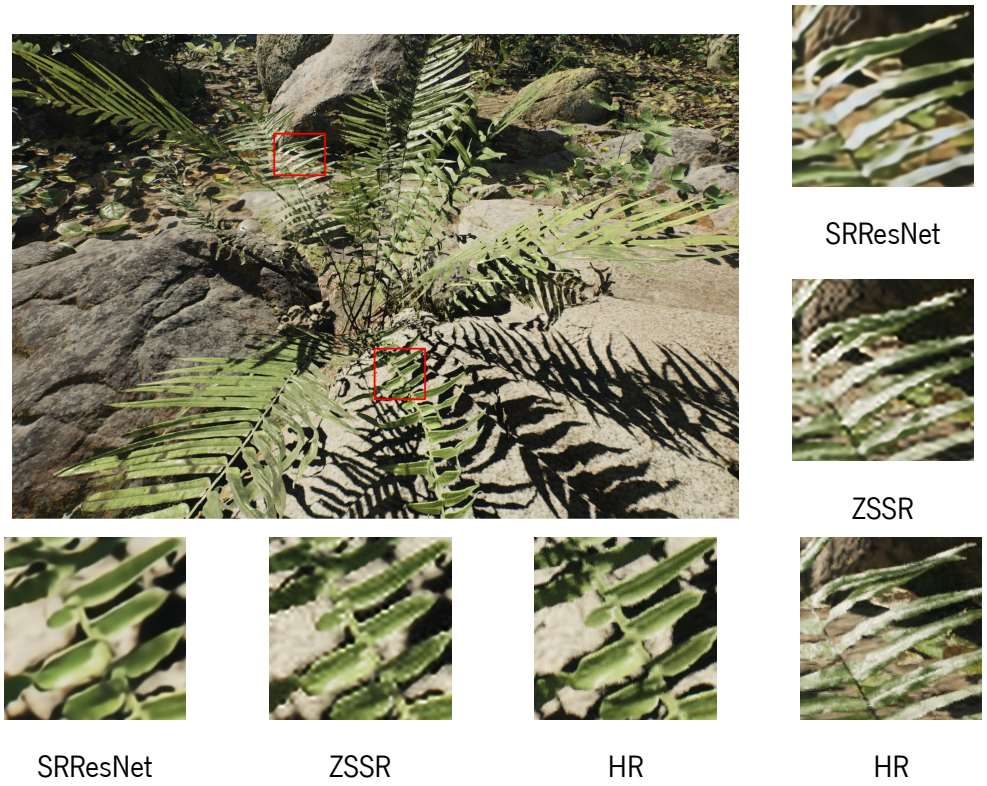


Figure 31: Unreal SRResNet vs ZSSR, tested on a different image from the True LR test set.

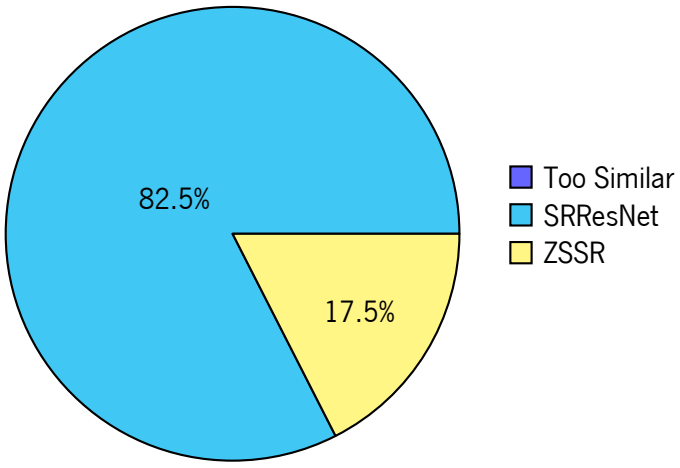


Figure 32: Perceptual Results for Figure 31.

Table 9: Results from ZSSR in a Forest test set

Test set	Progressive	**SSIM**	**PSNR**	**LPIPS**
Forest	Yes	0.7310	26.64	0.2335

4.3 Summary

After all the experiments performed we can take some conclusions on the results:

- Using the Domain dataset instead of the General one for SR on forest images has shown to be beneficial for both models. However, creating a specialized dataset using a synthetic scene did not result in a significant visual improvement for SRResNet and even caused SRGAN to produce more unrealistic outputs. The results from SRGAN model expressed a visual improvement with higher dataset cardinalities, different from SRResNet, where those visual differences were almost imperceptible.

- In most situations (General and Domain datasets) SRGAN generates the most detailed results. However, the resulting image has usually more differences to the original, as the model tries to fill the lost information. In the context of these networks, the further training of SRResNet into SRGAN is only advantageous if the final objective does not require the images to converge as close as possible to the originals, but only the images to be visual appealing.

- The tests on the True LR training set revealed that using interpolation methods to generate the LR images is a highly effective approach. The results even exceeded the models where the LR images were directly generated by the graphics engine. Apparently, downscaling methods can still offer great performance by providing more controlled and consistent training data.

- The unsupervised approach showed substantially lower performance compared to the supervised models on test sets where bicubic interpolation was used to generate LR images. However, ZSSR demonstrated a significant improvement when evaluated on the True LR dataset. In some cases, it managed to capture details missed by the other models. Despite this, it also led to an increase in aliasing artifacts. As a result, the supervised networks still delivered the best overall performance, maintaining a superior balance between detail preservation and artifact reduction.

- Pixel based losses (SRResNet) tend to yield higher SSIM and PSNR scores, while content based losses (SRGAN) perform better in terms of LPIPS. It was clear that these metrics tend to evaluate different features. SSIM and PSNR often fail to align with human visual perception, and although LPIPS better captures visual realism, it can also be misleading. In certain cases, LPIPS favors image sharpness and may prefer images with more artifacts. On the other hand, PSNR and SSIM favor the pixel level fidelity to the real image, even if those differences cant be noticed by humans, often giving preference for blurry results.

Chapter 5

Conclusion

Single Image Super Resolution (SISR) remains a challenging problem. Deep learning has significantly advanced the capabilities of SISR methods, providing richer high frequency detail, however, there is still a long road ahead.

This project has reviewed critical aspects of deep learning-based SISR, focusing on evaluation metrics, loss functions, training datasets, and the comparison between supervised and unsupervised approaches.

Loss functions directly influence the quality of the generated HR images. Our review highlights the relation between the metrics and the loss functions, with SSIM and PSNR providing better scores for pixel based losses, whereas LPIPS provides better scores for content-based losses. SSIM and PSNR are not good predictors of our human perception of image quality, as there were situations where the subjective study's results contradicted these metrics. LPIPS has shown better alignment with human perception in most situations, hinting at being a useful complement to the metrics commonly used for image quality assessment. Different metrics favor different aspects of images; hence, none should be used alone to measure SR results.

The specificity of the training set is also relevant, however while an improvement can be seen in particular results, it is not clear how much the specificity benefits the resulting images. Visually, it is clear that the highest level of specificity can even produce worse results. This result is somewhat counter intuitive, and it may be due to the selection of the forest domain or the use of synthetic data. Other domains and synthetic scenes may behave differently, and more research is required to draw a conclusive answer.

It was demonstrated that using interpolated methods to obtain LR images does not cause the models to simply learn the inverse of the interpolation process. Instead, the models are able to generalize beyond the specific downscaling method, capturing broader patterns and features that contribute to effective SR.

Our comparison of supervised and unsupervised SISR approaches reveals the distinct advantages and challenges associated with each method. Supervised approaches are based on large datasets of paired LR-HR images. Although these methods are capable of achieving higher performance, they require a con-

siderable amount of training time. Unsupervised approaches, on the other hand, do not require a training set, saving training time at the cost of larger inference time (which includes training for a single image). Unsupervised methods provide greater flexibility but often do not achieve the same balance between detail preservation and artifact reduction as their supervised counterparts. Furthermore, unsupervised methods are not feasible when considering a large number of images to upscale.

In conclusion, although the field of SISR has seen considerable progress, the metrics used to evaluate these methods are still perhaps the biggest issue in this area.

5.1 Future Work

In order to further support the results obtained and draw more conclusive answers on the influence of the datasets, other experiments are essential. Other image domains may behave differently and show more significant variations. The use of synthetic data has also raised the question of its suitability for SR tasks, highlighting the need for further experiments with different synthetic scenes to better understand its effectiveness and limitations. Even the true LR images used were obtained using a synthetic scene (generated by the graphics engine), so the models may not exhibit the same outcome with real world LR samples.

The models we used were selected mainly to represent different facets of the SISR current approaches. However, once the model architectures directly influence the final results, different architectures may exhibit different behaviors.

Finally, exploring or even developing other image quality assessment metrics with a better suitability for SR is crucial for the area's advancing as a whole.

Bibliography

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 126–135, 2017.

Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.

Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12299–12310, 2021.

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

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*, pages 184–199, 2014.

Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 391–407. Springer, 2016.

Azuma Fujimoto, Toru Ogawa, Kazuyoshi Yamamoto, Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Manga109 dataset and creation of metadata. In *Proceedings of the 1st international workshop on comics analysis, processing and understanding*, pages 1–5, 2016.

Epic Games. Electric dream environment, 2023. URL https://www.unrealengine.com/en-US/electric-dreams-environment.

Garas Gendy, Guanghui He, and Nabil Sabor. Lightweight image super-resolution based on deep learning: State-of-the-art and future directions. *Information Fusion*, 94:284–310, 2023.

Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1664–1673, 2018.

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

Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, 2010. doi: 10.1109/ICPR.2010.579.

Xuecai Hu, Haoyuan Mu, Xiangyu Zhang, Zilei Wang, Tieniu Tan, and Jian Sun. Meta-sr: A magnification-arbitrary network for super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1575–1584, 2019.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016.

Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 1981.

Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016.

Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In *Proceedings of the European conference on computer vision (ECCV)*, pages 517–532, 2018.

Juncheng Li, Faming Fang, Jiaqian Li, Kangfu Mei, and Guixu Zhang. Mdcn: Multi-scale dense cross network for image super-resolution. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(7):2547–2561, 2020.

Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3867–3876, 2019.

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.

António Ramires Fernandes Lucas Carvalho. Exploring super resolution deep learning approaches. In *to be published: ICGI 2024 digital proceedings and IEEE Xplore Digital Library*, 2024.

David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001.

Kangfu Mei, Aiwen Jiang, Juncheng Li, Bo Liu, Jihua Ye, and Mingwen Wang. Deep residual refining based pseudo-multi-frame network for effective single image super-resolution. *IET Image Processing*, 13(4): 591–599, 2019.

Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012.

Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 191–207. Springer, 2020.

Prachi R Rajarapollu and Vijay R Mankar. Bicubic interpolation algorithm implementation for image appearance enhancement. *Ijcst*, 8(2):23–26, 2017.

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016a.

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1874–1883, 2016b. doi: 10.1109/CVPR.2016.207.

Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3118–3126, 2018.

Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.

Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017a.

Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017b.

Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017.

Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE international conference on computer vision*, pages 4799–4807, 2017.

Todd Veldhuizen. Measures of image quality. URL https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/VELDHUIZEN/node18.html.

Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 606–615, 2018a.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018b.

Zhihao Wang, Jian Chen, and Steven C. H. Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3365–3387, 2021. doi: 10.1109/TPAMI.2020.2982166.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Qing Yan, Yi Xu, Xiaokang Yang, and Truong Q Nguyen. Single image superresolution based on gradient profile sharpness. *IEEE Transactions on Image Processing*, 24(10):3187–3202, 2015.

Wenming Yang, Xuechen Zhang, Yapeng Tian, Wei Wang, Jing-Hao Xue, and Qingmin Liao. Deep learning for single image super-resolution: A brief review. *IEEE Transactions on Multimedia*, 21(12):3106–3121, 2019.

Yuan Yuan, Siyuan Liu, Jiawei Zhang, Yongbing Zhang, Chao Dong, and Liang Lin. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 701–710, 2018.

Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on computer vision and pattern recognition*, pages 2528–2535. IEEE, 2010.

Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers 7*, pages 711–730. Springer, 2012.

Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018a.

Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018b.

Appendix A
ICGI 2024 Paper

Winner of the Best Paper Award; The International Conference on Graphics and Interaction. [Lucas Carvalho, 2024]

Exploring Loss Functions, Metrics, and Datasets for Super Resolution Deep Learning Models

Lucas Carvalho
Department of Informatics
Universidade do Minho
Braga, Portugal
pg50555@alunos.uminho.pt

António Ramires Fernandes
ALGORITMI Research Centre / LASI
Universidade do Minho
Braga, Portugal
ORCID:0000-0002-3680-572X

***Abstract*—Single Image Super Resolution (SISR) is an active area of research within the deep learning community, aiming at enhancing the resolution of a low-resolution (LR) observation to produce its high-resolution (HR) counterpart.**

We begin by examining various loss functions employed in SISR models, analyzing how they affect performance metrics commonly used to assess these methods. The evaluation of different metrics on the quality and accuracy of the generated HR images is critically evaluated using HR generated samples.

Then we explore the significance of the training datasets, namely how the specificity of these datasets impacts the performance of SISR models. Both quantitative metrics and qualitative assessments of the HR samples are used to gauge the effectiveness of different training datasets.

Finally, the paper compares two distinct SISR approaches: the supervised approach, where models are pre-trained on extensive datasets before inference, and the unsupervised approach, which generates HR images from a single LR image without prior training.

***Index Terms*—super-resolution, evaluation metrics, perceptual-based, pixel-based, deep leaning, dataset specificity**

I. Introduction

The demand for enhanced image resolution is present in areas such as medical imaging and satellite imagery. The goal behind Single Image Super Resolution (SISR) is to reconstruct a high-resolution (HR) image from a low-resolution (LR) observation, where it aims to recover the lost high-frequency details required for obtaining natural and/or informative HR images.

Prior to the entrance of deep learning methods in this area, interpolation methods like bicubic or Lanczos, often failed to produce convincing results, with results getting degraded as the upscaling factor goes up [1]. These methods tend to generate overly smooth images, lacking the high frequency details present in the original high-resolution scenes. In recent years, the deep learning community has developed methods to address the limitations of conventional SISR techniques, with Convolutional Neural Networks (CNNs) achieving some successes in capturing high frequency information required to create HR images [2].

This paper focuses on several relevant aspects that influence the effectiveness of these methods. First, we delve into the various loss functions utilized in training SISR models, and metrics commonly used to assess them. Loss functions guide the optimization process, and their choice significantly impacts the quality of the generated HR images. We examine common metrics to evaluate SISR methods, including both pixel based and content based metrics, with the support of visual inspection of generated HR images.

Next, we explore the influence of training datasets in the performance of SISR models. Different levels of specificity are used in this study. We investigate the impact of three datasets, with different levels of tailoring to a specific domain, on the quantitative metrics and visual quality of the reconstructed images.

Furthermore, the paper contrasts two distinct approaches to SISR: supervised and unsupervised methods. Supervised SISR relies on pre-training models with large datasets containing paired LR-HR images, where models learn a mapping from LR to HR images. On the other hand, unsupervised SISR does not require paired training data, instead leveraging on the power of the cross-scale internal recurrence of image-specific information.

II. State of the art

SR learning-based methods have gained significant attention for their high quality results. Among these, deep learning-based SISR algorithms have recently demonstrated superior results, outperforming traditional methods and other learning-based approaches [2]. This section provides some context in some crucial aspects of SISR, namely: the datasets, models and architectures, loss functions, and the image quality assessment methods commonly used to evaluate SISR models.

A. Super-Resolution Datasets

Today there are a variety of datasets available for SR, which greatly differ in size, quality, resolution, and diversity. Commonly, datasets only provide HR images and the LR images are usually obtained by bicubic interpolation with anti-aliasing. The most reported test datasets in the literature are (BSDS100 [3], General-100 [4], Set5 [5], Set14 [6] and Urban100 [7]).

Regarding training datasets, the DIV2K [8] is the most popular among the SR community. However, the optimal size of the training dataset remains a question. Some researchers rely only in the DIV2k [9], [10] while others combine it with

other datasets, such as OutdoorScene [11] and Flickr2K [12] or even the ImageNet database [13] to build massive datasets [14], [15]. During training the models use small crops as samples, while evaluation can be performed at any resolution.

B. SR Strategies and Architectures

Network design and architecture advancements are of great importance in deep learning. For SR, researchers have proposed several different designs and strategies.

The pioneer deep learning model for SR was proposed as Super Resolution Convolutional Neural Network (SRCNN) [2], a three layer CNN that approximates the complex mapping between the LR and HR spaces. This method immediately showed vast superiority over the conventional approaches and motivated the blooming of CNN based SR methods.

Residual learning was proposed as a way of easing the training of very deep networks, and was latter applied in SR [14]. As LR and HR images share much information, rather than learning the complete mapping, the network focuses on modeling the difference or residual image between these LR and HR images. Residual learning is one of the most popular strategies for SR, not only facilitating the training of very deep networks but also contributing to the design of lightweight models that still achieve good results [19].

In **Recursive learning**, the strategy is based on the same module being updated on a iterative way, minimizing the parameters. Despite reducing the parameters, recursive learning networks require higher computational power and too many stacked layers may cause the problem of vanishing/exploding gradient , that makes it often used in combination with other strategies that help preventing this problem, such as residual learning [20].

Dense connections lies in each layer receiving input from all preceding layers. The connections form short paths between most layers addressing issues like vanishing or exploding gradients and helps the flow of deep information across layers. Dense connection mechanism was introduced in SISR [21] not only employing dense connections at the layer level but extending this concept to the block level. Here, the output of each dense block is interconnected through dense connections, which allows the use of low-level and high-level features for the prediction.

Unsupervised learning has been the least explored approach in SR. However, in recent years, a new technique called Zero Shot Super Resolution (ZSSR) [22] has been proposed to generate SR images without relying on a pre-existing training dataset. ZSSR operates using a single LR image to directly train a deep learning model using image augmentation techniques. This allows the model to learn directly from the internal features and patterns within the image itself.

In addition to these approaches, numerous other designs have been proposed. Many of these models tend to combine these foundational approaches with new, innovative techniques to further enhance performance and efficiency [19]. For example, curriculum learning [23] involves gradually increasing the complexity of the learning task, which helps reducing training difficulty and enhances overall performance. Multi-scale learning [24] aims to make full use of the different characteristics that images may exhibit at different scales. The continuous evolution and hybridization of these strategies highlight the dynamic and rapidly advancing field of super-resolution network architectures.

C. Loss Functions

In SR the loss function role is to guide the iterative optimization process of the model by computing some error between the SR and the HR image. There are two main categories of loss functions and each one aims for different aspects of the images.

Pixel loss is the simplest and most popular type of loss function in this field [2], [14], [20], [22]. It provides a straightforward method to evaluate the disparity between two images at the pixel level, aiming to minimize this difference and bring the images as close as possible. Pixel loss is commonly referred to as L2 loss or L1 loss, which correspond to mean square error (MSE) and mean absolute error (MAE), respectively. As the most used IQA methods are highly correlated with pixel differences, such as PSNR, it is common for pixel loss models to achieve the best scores with objective methods.

Content loss [25] uses a pre-trained classification network to measure the semantic difference between images, and can be further expressed as the Euclidean distance between the high-level representations of two images. Adversarial loss in SR, is also a form of content loss [14]. Here the SR images are obtained with a Generative Adversarial Network (GAN). The generator job is to generate HR images, while the discriminator is used to determine the authenticity of the generated samples. The loss function is usually computed as a weighted sum of a content loss and an adversarial loss component.

D. Image Quality Assessment Methods

Image quality assessment (IQA) methods can be characterized into subjective methods (human perception of if an image looks natural and has good quality) and objective methods (quantitative methods by which image quality can be numerically computed).

Subjective methods can provide valuable insigths regarding the quality obtained. On the other hand these methods are time-consuming and usually expensive. Hence, objective methods are currently mainstream. Each approach provides different perspectives regarding the resulting image, thus they are not necessarily consistent among different approaches. Objective methods are usually unable to capture the human visual perception very accurately, which may lead to ambiguous results.

Peak Signal-to-Noise Ratio (PSNR) [16] is one of the most popular reconstruction quality measurement of lossy transformation, particularly in SR. This metric can sometimes be misleading [16] as the evaluated image might not be visually similar to the reference image. This happens because it only focuses in the differences between corresponding pixels

instead of visual perception. In order to consider other aspects of the images, the Structural Similarity Index metric (SSIM) was proposed [17]. It aims to compare the contrast, luminance, and structural details within the images. The current mainstream is to use both of these metrics to measure the SR.

Another metric, Learned Perceptual Image Patch Similarity (LPIPS) [18] evaluates the perceptual similarity between two images by considering how humans notice the differences between them. Unlike traditional metrics, which primarily focus on pixel-level comparisons, LPIPS uses deep convolutional neural networks (CNNs) to capture and compare high-level visual features. By passing the images through pre-trained CNNs. LPIPS computes the distance between the feature representations of image patches.

However, none of the quantitative methods can accurately simulate human perception. This is why subjective methods like collecting peoples opinion, known as Mean Opinion Score (MOS) is used by some researchers [14].

III. Method and Contribution

The work described in here aims to explore some issues related to SISR:

- Evaluating the benefits of having specific datasets. Three datasets were built for this purpose. One consisting on a combination of super resolution datasets commonly used for training SR models. Secondly, a Domain dataset containing only forest related images, and finally a scene specific dataset of a single forest created in Unreal Engine 5.
- Evaluating metrics, their relation to the loss functions, and the consistency amongst them. For this purpose two models were trained with different loss functions (perceptual loss and pixel loss).
- Evaluate the impact of the cardinality of the training set.
- Comparing supervised and unsupervised methods both regarding output quality and other factors as the training effort and evaluation effort.

A. Datasets

For this study, training was performed on three distinct datasets with different levels of specificity: a general dataset, a domain specific dataset with real images, and an even more specific synthetic dataset created in Unreal Engine.

Furthermore, to explore the impact of the cardinality of the datasets, each version had 5 different cardinalities, totaling 15 different training datasets.

Based on the DIV2K dataset, which contains 800 images, this was set as the smallest cardinality. The remaining cardinalities were obtained doubling iteratively the number of samples, resulting in cardinalities: 800, 1600, 3200, 6400, and 12800 images.

For the general domain dataset, in the smallest cardinality, 800 images, it is identical to DIV2K. Images from other datasets (Flick2K, OutdoorScene), and images retrieved from Flickr, were added to fulfil the larger cardinality.

For the specific domain the forest theme was chosen mainly for its high level of detail, as well as for the abundance of images. Images for these datasets were retrieved from Flickr, subsequently filtering the ones considered blurred to ensure the quality of the dataset.

For the more specific dataset a synthetic forest environment was created in Unreal (Electric Dreams Environment [26]). Images were captured from random camera positions above the ground, and were subsequently reviewed to eliminate any possible major occlusions with objects.

To facilitate the distinction between the datasets, these will be referred henceforth as ”General”, ”Domain”, and ”Unreal” in increasing order of specificity.

All the datasets can be found here:
https://github.com/lucasCarvalho64/Datasets-for-Super-Resolution.git

B. Models

SR models have improved over time as new strategies and deeper networks are proposed. These improvements are usually accompanied by higher computational demands and increased training times. The purpose here is not to evaluate the models themselves, hence, the models used in here were selected mainly for representing different facets of the SISR current approaches.

Super Resolution GAN (SRGAN) [14], proposed a generative adversarial network with the objective of creating more perceptually realistic images, in which the generator was referred to as Super Resolution Residual Network (SRResNet). In SRGAN, the SRResNet is trained with the goal of fooling a discriminator that learns to distinguish super-resolved images from real HR images. The Generator is composed of 16 residual blocks. The residual block has become the basic unit in these network structure consisting of two 3×3 convolutional layers, two batch normalization layers, and a ReLU activation function in between.

For the training process, in [14] the authors first train only the SRResNet with MSE as the loss function and employ it as initialization for the SRGAN's generator, this way the discriminator receives proper SR images from start. The SRGAN loss function, as mentioned above, was formulated as the weighted sum of a content loss and an adversarial loss component. Here, we followed the same training strategy, resulting in a pixel-based model (SRResNet) and a perceptual-based model (SRGAN).

For more detailed information about these networks, we recommend reading the original paper [14], as our work follows the implementation described therein.

Supervised SR networks trained on large and diverse collections of LR and HR image examples aim to capture the vast variety of all possible LR-HR relationships. Consequently, these networks tend to be extremely deep and complex. In contrast, the diversity of LR-HR relationships within a single image is much lower, allowing for encoding with a much smaller architecture in a simpler image-specific model. This is the idea behind ZSSR [22].

The network has 8 hidden layers, each containing 64 channels and ReLU activations. Similarly to the supervised approach, it only learns the residual between the interpolated LR and its HR correspondence.

IV. Evaluation using Supervised Methods

For each domain we conducted five training runs, starting with 800 images and doubling the number each run up to 12800 images, on SRResNet and SRGAN, resulting in a total of 30 models. We trained all networks on a NVIDIA RTX 3060 with a learning rate of 10^{-4} and a batch size of 16, using 192x192 high-resolution (HR) images, always for a scale factor of 4. The low-resolution (LR) images were generated by downsampling the HR images using bicubic interpolation with a downsampling factor of 4. The SRResNet networks were trained for 10^6 update steps and served as an initialization for the SRGAN networks, which were further trained for 10^5 update steps.

All models were evaluated on two distinct test sets. The first dataset consists of a sample on 10 images from the Unreal forest dataset, different from the images used during training. The second is the Set14 [6], a general benchmark test dataset with fourteen images widely-used for SR tasks. We used the mainstream IQA methods, PSNR and SSIM, in combination with LPIPS , as well as some result samples to visually compare the results.

A. General Dataset

Table I presents the results of models trained with the General dataset. Regarding the metrics, results show a relative agreement between all the metrics when considering the Set14 test set. Results tend to get better as the number of samples in the dataset increases, although there are no significant improvements.

Considering the Unreal test set, the differences as the dataset increases are more significant for the SRGAN. Furthermore, for the Unreal test set we can observe a discrepancy between the metrics as the dataset increases, with LPIPS tending to get worse results, and both SSIM and PSNR improving. Nevertheless, the only significant improvement that can be observed in this test set is for the model trained with the SRGAN, with the SSIM metric going from 0.6492 to 0.6887.

Since we are testing in a specific domain, and taking into account that the General datasets do not have a significant number of forest images, it seems reasonable to expect an increase in performance as the number of samples in the dataset goes up.

Another interesting take is that even when considering the smaller datasets, the SRResNet model achieves a higher SSIM score than SRGAN. This is to be expected as SRResNet uses pixel based loss, whereas SRGAN uses content based loss. These results are in line with the results presented in [14] where SRResNet also obtains a higher SSIM and PSNR score than SRGAN. Worth noting is that in [14] the authors also use Mean Opinion Score (MOS) as an evaluation metric. The reported results indicate that MOS is not aligned with SSIM or PSNR, with SRGAN having a higher MOS score. Our results show the same tendency, with SRGAN having a better LPIPS score than SRResNet, hinting at LPIPS higher affinity with human perception.

TABLE I: Models trained with the General dataset

Model	Test set	Size	SSIM	PSNR	LPIPS
SRResNet	**Set14**	800	0.7906	27.94	0.2102
		1600	0.7971	28.25	0.2122
		3200	0.7995	28.31	0.2119
		6400	0.8026	28.48	0.2107
		12800	**0.8046**	**28.67**	**0.2098**
	Unreal	800	0.6906	26.60	0.2271
		1600	0.6943	26.74	0.2330
		3200	0.6779	26.65	**0.2238**
		6400	0.6991	26.92	0.2352
		12800	**0.7007**	**27.00**	0.2345
SRGAN	**Set14**	800	0.7520	27.10	0.1774
		1600	0.7479	27.14	0.1699
		3200	0.7453	27.05	**0.1613**
		6400	0.7598	27.28	0.1651
		12800	**0.7698**	**27.88**	0.1644
	Unreal	800	0.6492	25.99	**0.2120**
		1600	0.6609	26.20	0.2133
		3200	0.6779	26.64	0.2238
		6400	0.6673	26.34	0.2302
		12800	**0.6887**	**26.92**	0.2347

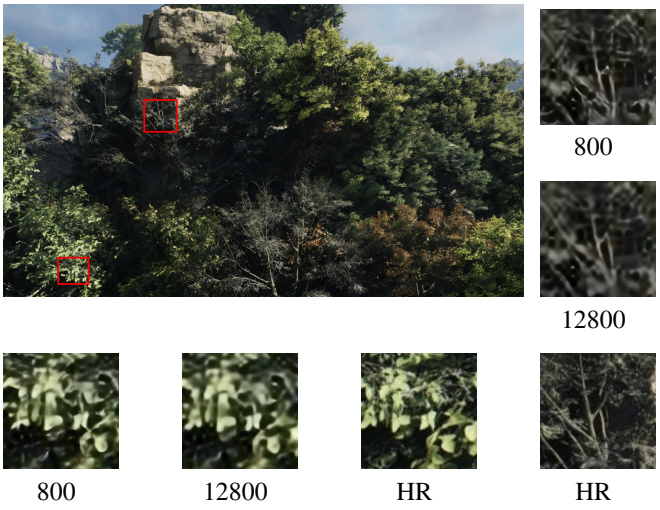


Fig. 1: General SRResNet results with 800 and 12800 training images from Unreal dataset image.

Fig. 1 shows an evaluation HR sample with closeup regions that can shed some light on this issue. The model trained with the highest cardinality dataset provides results that look blurrier than when training with the smallest dataset. On the other hand, in particular when looking at the region with branches, the sharper image also has clearly visible artifacts. SSIM and PSNR do rate the 12800 dataset images higher, whereas LPIPS considers them worse. This example clearly shows that different metrics evaluate different features, and hence none should be considered in isolation as an absolute measure of image quality in the context of SR methods.

B. Domain Dataset

Table II presents the results of the models trained with the Domain dataset. As expected, due to the specificity of the training dataset, the results on Set14 suffer significantly when compared with models trained with the General dataset.

For the Unreal dataset it can be observed that the SRGAN's results for SSIM and PSNR suffer a significant reduction with increasing dataset sizes, apparently overfitting the training data. Note that the SRGAN includes the SRResNet, and further trains it. This extended training could lead to overfitting.

Tests performed on the Unreal test set, when comparing with the models trained with the General dataset, show an improvement in both PSNR and SSIM.

Again, there are some discrepancies amongst the metrics. For instance considering the SRGAN tested with the Unreal dataset, LPIPS shows a considerable improvement as the training dataset increases, whereas SSIM and PSNR get significantly poorer results.

Fig. 2 illustrates the progressive changes in the same HR image produced by SRResNet as the size of the training dataset increases. We notice that the results tend to become slightly blurred, but the smaller branches look more defined from cardinality 1600 forward. These contrasting changes do not allow a definitive statement on what image looks better, in particular because the differences are very subtle. However, it highlights that pixel-wise metrics and perceptual metrics analyse different important aspects of the images. Considering that LPIPS tends to penalize blur, the visual inspection is in line with the scores obtained for this metric.

In Fig. 3 we can see how the result varies with the training set size for SRGAN. From 800 to 3200, image sharpness increases, but more artifacts are also present in 3200 dataset. At 6400 images, the results seem blurrier again, while getting sharper again with the 12800 images. In this case this last sample seems clearly the most natural. SSIM and PSNR metrics tell a different story with the model trained with the 800 dataset getting significantly higher scores. Only the LPIPS metric is in accordance with the visual inspection.

TABLE II: Models trained with the Domain Dataset

Model	Test set	Size	SSIM	PSNR	LPIPS
SRResNet	**Set14**	800	0.7358	26.79	**0.2762**
		1600	0.7461	27.09	0.2876
		3200	0.7475	27.21	0.2912
		6400	0.7506	27.37	0.2962
		12800	**0.7539**	**27.50**	0.2936
	Unreal	800	0.7164	27.22	**0.2287**
		1600	0.7223	27.37	0.2341
		3200	0.7247	27.50	0.2382
		6400	0.7268	27.59	0.2447
		12800	**0.7301**	**27.65**	0.2422
SRGAN	**Set14**	800	**0.7241**	**26.54**	0.3058
		1600	0.7143	26.41	**0.2450**
		3200	0.6613	25.32	0.2453
		6400	0.6560	25.06	0.2704
		12800	0.5943	23.86	0.2851
	Unreal	800	**0.6972**	**26.80**	0.2540
		1600	0.6760	26.72	0.2266
		3200	0.6136	25.51	0.2100
		6400	0.5982	25.60	0.2550
		12800	0.6301	25.96	**0.2074**

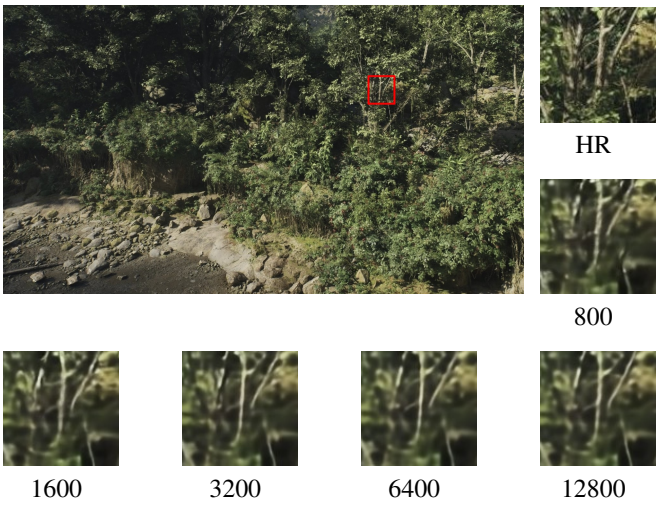


Fig. 2: Domain SRResNet from 800 to 12800 training images tested on Unreal test set image.

C. Unreal dataset

Table III lists the results of all models trained with the most specific dataset, the Unreal dataset. Overall, the metrics behaviour for Set14 don't differ significantly from the results obtained with the Domain dataset.

Considering the Unreal test set there is a slight improvement when testing with SRResNet. However, when testing with the SRGAN model, the best results for the Unreal test set are worse than those obtained with models trained with the general and domain training sets. Still, the best LPIPS score for this test data set is obtained with this training set.

Looking at HR samples generated with these models can provide further insight into the results. Fig. 4 and Fig. 5 show samples generated with each of the models. In Fig. 4 it can be observed that the upsampled images obtained with SRResNet models trained on Unreal and Domain datasets are almost identical. which aligns considerably with the numeric data. Fig. 5, showcasing upsampling using the SRGAN, tells a different story. On the one hand, for the 12800 case, the metrics show an improvement in all metrics when using the Unreal vs the Domain training set. However, the figure clearly shows severe artifacts when considering the Unreal training set. Furthermore, the Domain samples show more detail.

This example clearly raises the issue of the suitability of these metrics to independently evaluate quality of a SR model.

V. Unsupervised Approach

While the methods in the previous section required a training dataset, ZSSR [22] takes as its only input the image to be upsampled, without any prior training. Since the training set consists of only the image to be upsampled, data augmentation is used to create more LR-HR example pairs by generating multiple versions of the input image at lower resolutions. Additionally, the training set is enhanced by applying four

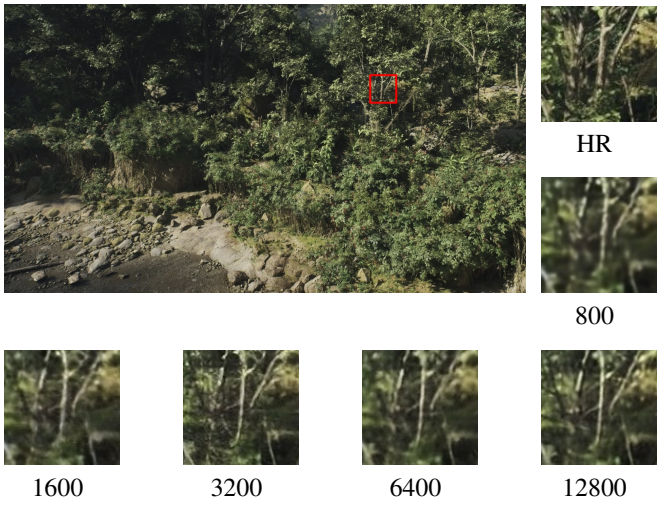


Fig. 3: Domain SRGAN from 800 to 12800 training images tested on Unreal test set image.

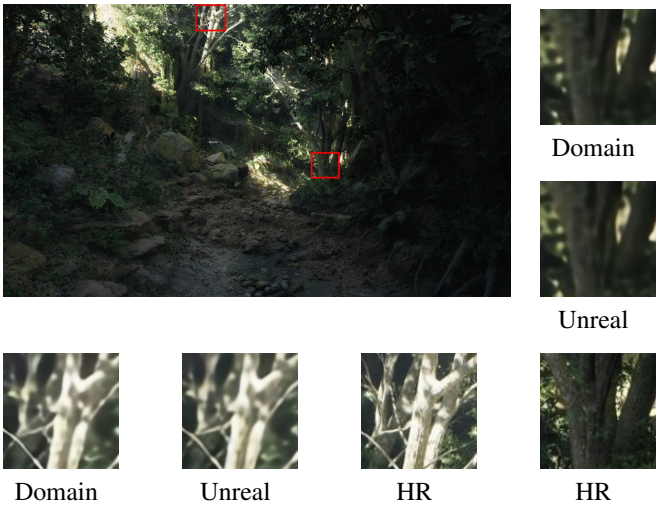


Fig. 4: Domain vs Unreal SRResNet results with 12800 training images from the same image from the Unreal test dataset.

TABLE III: Models trained with the Unreal Dataset

Model	Test set	Size	SSIM	PSNR	LPIPS
SRResNet	**Set14**	800	0.7078	26.19	**0.3006**
		1600	0.7141	26.46	0.3024
		3200	0.7305	26.63	0.3073
		6400	0.7313	26.78	0.3132
		12800	**0.7446**	**27.10**	0.3156
	Unreal	800	0.7157	27.18	**0.2249**
		1600	0.7230	27.38	0.2323
		3200	0.7280	27.52	0.2369
		6400	0.7324	27.65	0.2383
		12800	**0.7348**	**27.74**	0.2411
SRGAN	**Set14**	800	0.6726	25.26	0.3227
		1600	**0.7372**	**26.91**	0.3578
		3200	0.7002	25.20	0.3274
		6400	0.6793	25.57	**0.2710**
		12800	0.7074	26.07	0.2851
	Unreal	800	0.6412	25.71	0.2388
		1600	0.6613	26.57	0.2173
		3200	0.6327	25.73	0.2170
		6400	0.6543	26.27	0.2067
		12800	**0.6764**	**26.61**	**0.2053**

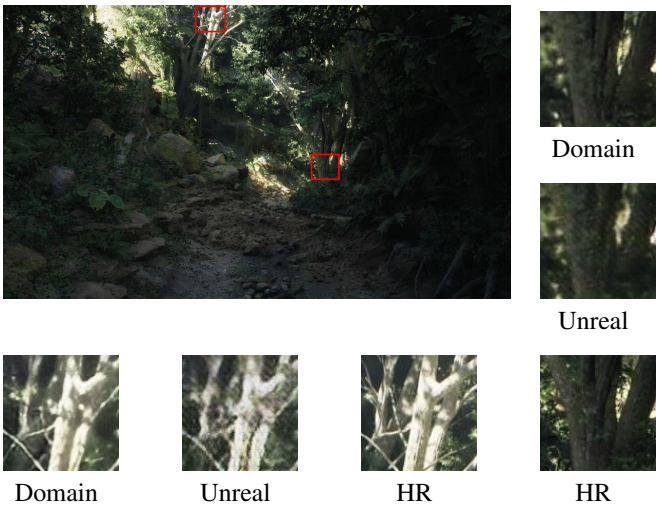


Fig. 5: Domain vs Unreal SRGAN results with 12800 training images from the same image from the Unreal test dataset.

90° rotations combined with vertical and horizontal mirror reflections. significantly increasing the number of specific examples derived from the same image.

Lastly, it uses a method similar to a self-ensemble, it generates 8 different outputs for the 8 rotations and flips of the test image and combines them. It is further combined with the back-projection technique, so that each of the 8 output images undergoes several iterations of back-projection and finally the median image is corrected by back-projection as well.

Learning rate starts on 0.001 and training stops when it reaches 10^{-6}. A linear fit of the reconstruction error is periodically taken. If the standard deviation is greater by a factor of 10 than the slope of the linear fit we divide the learning rate by 10.

As detailed in [22], ZSSR obtains better results when using a gradual increase in resolution. However, a gradual increase using 5 intermediate scale-factors also increased the runtime from 1 minute to 6 minutes per image making this approach unfeasible to process large quantities of images. While SRResNet takes about 12 hours to train, and another 10 hours for SRGAN, once trained it takes close to one second to generate one SR image.

A. *Results*

Table IV presents the results of the two ZSSR variants for a scale factor of 4. Based in these quantitative metrics, although there is an overall improvement using the progressive approach, it is almost negligible. However, a visual inspection of an upsample performed with the two versions of ZSSR, see Fig. 6, clearly shows that the results of the progressive approach have less artefacts, and better quality.

TABLE IV: Results from ZSSR

Test set	Progressive	SSIM	PSNR	LPIPS
Unreal	No	0.6861	26.19	0.2613
	Yes	**0.6886**	**26.48**	**0.2589**
Set14	No	0.7712	27.26	**0.2386**
	Yes	**0.7791**	**27.40**	0.2391

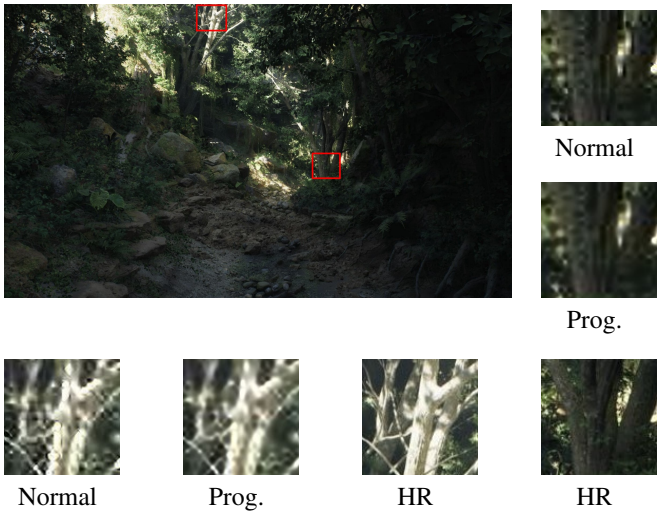


Fig. 6: Normal vs Gradual ZSSR results from the same image of the unreal test dataset.

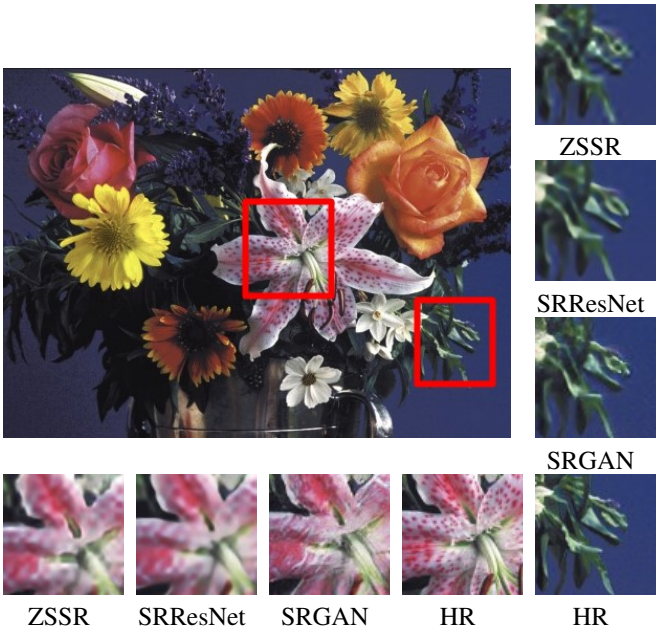


Fig. 7: ZSSR vs general SRResNet results with 12800 training images on Set14 image.

TABLE V: Results from ZSSR in a Forest test set

Test set	Progressive	SSIM	PSNR	LPIPS
Forest	Yes	0.7310	26.64	0.2335

When we compare the quantitative results for Set14 against the supervised models, we can observe that it is only clearly surpassed by SRResNet trained with the General dataset. Fig. 7 shows the upsampling result for all models in a Set14 image. Although it is impressive that ZSSR achieves a very good result based on a single image, SRResNet achieves slightly more defined results. SRGAN produces the crispiest result, although the pigmentation in the petals is clearly off target.

A result that stands out is the difference in the results for the two test sets. In order to further explore this issue a third test set was created, this time with 10 images from the Domain dataset. The results for this test set, presented in Table V, show a difference to the other two test sets, sitting in the middle for both PSNR and SSIM. Results for LPIPS are closer to Set14 than to Unreal. The fact that ZSSR performs better with images from real forests than Unreal generated ones, might be significant.

According to [22], their method "leverages on the power of the cross-scale internal recurrence of image-specific information". Can it be the case that forest generated in Unreal share less of this property than real forests? Are synthetic forest too random? These are interesting research questions regarding the procedural modelling of forest like scenarios, but lying outside the scope of this paper.

VI. Conclusion and Future Work

Single Image Super Resolution (SISR) remains a challenging problem. Deep learning has significantly advanced the capabilities of SISR methods, providing richer high frequency detail, however there is still a long road ahead.

This paper has reviewed critical aspects of deep learning-based SISR, focusing on evaluation metrics, training datasets, and the comparison between supervised and unsupervised approaches.

Loss functions directly influence the quality of the generated high-resolution images. Our review highlights the relation between the metrics and the loss functions, with SSIM and PSNR providing better scores for pixel based losses, whereas LPIPS provides better scores for content base losses. SSIM and PNSR are not good predictors of our human perception of image quality.

The specificity of the training set is also relevant, however while an improvement can be seen in particular results, it is not clear how much the specificity benefits the resulting images. Visually it is clear that the highest level of specificity can even produce worse results. This result is somewhat counter intuitive, and it may be due to the selection of the forest domain or the use of synthetic data. Other domains and synthetic scenes may behave differently, and more research is required to draw a conclusive answer.

Our comparison of supervised and unsupervised SISR approaches reveals distinct advantages and challenges associated with each method. Supervised approaches rely on large datasets of paired LR-HR images. Although these methods are able to achieve higher performance they require a considerable

amount of training time. Unsupervised approaches, on the other hand, do not require a training set, saving the training time, at the cost of larger inference time (which includes training for a single image). Unsupervised methods provide greater flexibility but often do not achieve the same level of detail and accuracy as their supervised counterparts. Furthermore, unsupervised methods are not feasible when considering a large number of images to upscale.

In conclusion, although the field of SISR has seen considerable progress, the metrics used to evaluate these methods are still perhaps the biggest issue in this area.

ACKNOWLEDGEMENTS

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

REFERENCES

[1] W. Yang, X. Zhang, Y. Tian, W. Wang, J. -H. Xue and Q. Liao, "Deep Learning for Single Image Super-Resolution: A Brief Review," in IEEE Transactions on Multimedia, vol. 21, no. 12, pp. 3106-3121, Dec. 2019

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13, pages 184–199. Springer, 2014.

[3] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, volume 2, pages 416–423. IEEE, 2001.

[4] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pages 391–407. Springer, 2016.

[5] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single image super-resolution based on nonnegative neighbor embedding. 2012.

[6] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers 7, pages 711–730. Springer, 2012.

[7] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed selfexemplars. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5197–5206, 2015.

[8] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 126–135, 2017.

[9] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 136–144, 2017.

[10] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16, pages 191–207. Springer, 2020.

[11] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super- resolution by deep spatial feature transform. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 606–615, 2018a

[12] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 114–125, 2017.

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

[14] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4681–4690, 2017.

[15] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12299–12310, 2021.

[16] Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In 2010 20th International Conference on Pattern Recognition, pages 2366–2369, 2010. doi: 10.1109/ICPR.2010.579.

[17] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4):600–612, 2004.

[18] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 586–595, 2018.

[19] Garas Gendy, Guanghui He, and Nabil Sabor. Lightweight image super-resolution based on deep learning: State-of-the-art and future directions. Information Fusion, 94:284–310, 2023.

[20] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In Proceedings of the IEEE international conference on computer vision, pages 4539–4547, 2017.

[21] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In Proceedings of the IEEE international conference on computer vision, pages 4799–4807, 2017.

[22] Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3118–3126, 2018.

[23] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 3867–3876, 2019.

[24] Juncheng Li, Faming Fang, Jiaqian Li, Kangfu Mei, and Guixu Zhang. Mdcn: Multi-scale dense cross network for image super-resolution. IEEE Transactions on Circuits and Systems for Video Technology, 31(7):2547–2561, 2020.

[25] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and superresolution. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pages 694–711. Springer, 2016.

[26] Epic Games. (2023). Electric Dream Environment. Retrieved from https://www.unrealengine.com/en-US/electric-dreams-environment