Super-resolution GANs for bone imaging

Claire Robin

Under the supervision of
Dr. Ievgen Redko & Dr. Marc Sebban,
Data Intelligence team,
at Hubert Curien Laboratory, Saint-Etienne

Machine Learning and Data Mining master degree, first year
Jean Monnet University, Saint Etienne
Spring 2021

Contents

Abstract

The low resolution of medical imaging is a major issue in clinical practice, especially in the context of bone imaging where the bone micro-structure – an important factor of bone strength – can be observed only at high resolution. So, reconstructing the high-resolution (HR) image from the low-resolution (LR) image is an important challenge, which allows a sooner and better detection of bone diseases, such as, for instance, osteoporosis. The current state-of-the-art in super-resolution (SR) for natural images are deep learning-based methods and applying them to the medical images with their multiple particularities requires specializing the design of such SR models in order to obtain significant improvements. This report explores several ways for improvement of the well-established super-resolution model called SRGAN. First, we explore the benefits of attention mechanisms with Squeeze-And-Excitation blocks and conclude that, contrary to natural imaging, it brings no significant gain in case of bone imaging. For the case of high-scale factors, we propose to use intermediate representation to improve the learning capacity of the models, without obtaining convincing results. Finally, we use the correlation of the clinical micro-structural bone measures between the original and the super-resolved dataset to evaluate the improvement of the super-resolution on the bone micro-structure reconstruction. The lack of data prevents to conclude to a strong correlation between the high resolution (HR) dataset and the super-resolved (SR) dataset. Images, codes, and results are available in this Github Repository.

1 Introduction

The low resolution is a key issue in medical images. In the context of osteoporosis, a disease characterized by the reduction of the bone mineral density (BMD, the ratio of weight to the volume of the bones) and an increased risk of fracture, the low accuracy of measurement limits its recognition. This leads to suboptimal medical management of patients, decreases the quality of life and life expectancy, and increases medical costs.

However, the latest research suggests that the micro-architectural quality of the trabecular bone is an important factor in bone strength and fracture risk. Several algorithms have been implemented to compute accurate measures of micro-architecture from bones images [27], [22]. These measures of micro-architecture, especially the mean trabecular thickness (Tb.Th), the mean trabecular spacing (Tb.Sp), trabecular network area density (Tb.NA), and marrow spacing are of significant interest for early diagnosis of osteoporosis [22], [10]. Nevertheless, the measurement of the density is highly dependent on the resolution, which allows revealing the details of microstructure. Unfortunately, due to the limitation of materials, slow scan speed, and radiation exposure, the medical images are often in a low resolution.

To tackle this problem we can use Super-Resolution (SR) methods, which reconstruct the high-resolution image from its low-resolution image counter-part. This solution has already been explored by You and al. (2019) [40] and Guha et al. (2020) [10]. The above-mentioned way is promising since medical images have multiple particularities (great regularity in the images, grayscale, 3D structure, very low resolution), suggesting that significant improvements of medical image reconstruction quality can be obtained by specializing the design of SR models.

This internship took place in the context of Rehan Jhuboo's Ph.D. thesis carried out in the Data Intelligence team of the Hubert Curien Laboratory in collaboration with SAINBIOSE, a laboratory of INSERM specialized in chronic and aging diseases of the vascular and osteoarticular systems. The objective of his thesis is to use the bone monitoring of astronauts undergoing accelerated aging to better understand and predict bone aging.

During this internship, we tried several ways of improvement based on a super-resolution baseline SRGAN [19]. First, we studied different loss functions; second, we added squeeze-and-excitation blocks commonly used as attention mechanisms in computer vision [13]. We also proposed 2 ways to improve the results in the case of high scale resolution factors by using intermediate representation. Finally, we validated the improvement of the super-resolution for the study of the bone micro-structure by computing clinically significant microstructural measures. Specifically, we compute trabecular microstructural measures such as thickness, spacing, number of trabecular from the super-resolved (SR) image, the low resolution (LR) image, and the high resolution (HR) image. Unfortunately, the lack of data does not allow us to conclude that there is a stronger correlation between SR and HR than between LR and HR.

We start by presenting some previous works on super-resolution and its application in the medical imaging in section 2. Then, we present the theory of GAN, the architecture of SRGAN, the architecture of Squeeze-And-Excitation (SE) block, and the metrics of image quality assessment in section 3. Lastly, we present our contribution and the results in section 4.

2 Related work

2.1 Image super resolution

The single image super-resolution (SISR) refers to the ill-posed problem of constructing a high-resolution image (I^{HR}) from its corresponding low-resolution image (I^{LR}).

In practice, the ground-truth I^{HR} is assumed to undergo an unknown degradation leading to I^{LR} and the latter may be caused by its downscaling, adding noise, blur, or other artifacts. The SISR gathers all the methods allowing to learn the inverse function of degradation ϕ^{-1} such that $I^{SR} = \phi(I^{LR})$ is a approximation of I^{HR} reconstructed from I^{LR}.

Each pixel in low resolution is coded by r^2 pixels or more in high resolution, with r being the scale factor. So, several reconstructions are possible for each low resolution, and an error in low resolution can be encoded by r^2 pixels in high resolution according to the reconstruction function. This means that small perturbation in the low-resolution data potentially leads to a large error in the reconstruction.

Image super-resolution is used in various computer vision applications: security [25], image recognition [5] and medical imaging [9], [47] to name a few. It is extensively studied and several methods were proposed this last decade with deep learning methods outperforming the shallow models on various SR benchmarks in the recent years [1], [37].

The first great improvement in SR deep learning method occurred in 2015 with SRCNN [3], a convolutional neural network (CNN) able to learn an end-to-end mapping between the LR and the HR image. In their paper, the authors also demonstrate the strong relationship between a convolutional neural network (CNN) and the sparse-coding SR methods (eg. [39]). This is the previous most popular method in SR, which roughly generates an SR image using a low-resolution dictionary and a high-resolution dictionary. Then, several deep learning-based methods has been proposed following the latest advances in deep learning in order to increase the quality of the super-resolution and the stability of learning, reduce the number of parameters and the learning time, and the ability to generalize to new datasets.

In parallel with the CNN-based method, the GAN-based methods are vastly explored. Wang and al. (2018) [33] proposed a super-resolution conditional GAN-based on Pix2pix [15]; SRGAN [19] was designed based on a GAN for super-resolution using a perceptual loss, and ESRGAN [34], the update of SRGAN, improved the result of the latter by using Residual-in-Residual Dense Blocks and relativistic GAN.

The super-resolution is a large area extensively explored this last decade, and all these methods have been proposed and tested on natural image datasets. The medical imaging, however, has some particularities (great regularities of a dataset, small image, small dataset, non-paired images), so a lot of research is try-

ing to adapt super-resolution models to medical images.

2.2 Super-resolution for medical imaging

Super-resolution was extensively used for medical imaging [20]. For instance, CSR-GAN ([47]) combines a conditional GAN (C-GAN) with SRGAN [19] by using differential geometric information, the Jacobian determinant (JD), and curl vector (CV) as conditional input of both the discriminator and generator of SRGAN and they use a content loss based on the CV feature maps. Jiang and al. (2020) [16] propose an improvement of SRGAN by using dilated residual networks [42]: a stack of residuals blocks based on a dilated convolutional layer [41], which allows having no reduction of output size without improving the number of parameters. You and al. (2018) [44] reach the state of the art with a low computational cost by using GAN-CIRCLE [40]: a super-resolution semi-supervised model on computed tomography (CT) images (ie. bone images). They improve their model with the Wasserstein distance to address the training problem of GAN, a new loss based on joint constraints and they incorporate in their model a deep convolutional neural network (CNN), residual blocks, and network in networks [21], i.e parallel 1×1 CNN to compress the output of the hidden layer. Guha and al. (2020) [10] use GAN-CIRCLE too, without modification, however, they measure the improvement of the super-resolution with linear correlation, concordance correlation coefficients (CCC), and Bland-Altamard plot on trabecular density measures of LR, SR, and HR datasets. [38] propose a multimodal CT super-resolution method based on ESRGAN [34]. Wide-attention SRGAN (WA-SRGAN) ([30]) is a SRGAN model with wide residual block [43] instead of residual blocks. They pre-trained the VGG-loss on medical images since the original model has been trained on natural images. They add a self-attention layer [46] in both networks to capture the most important features, and a Wasserstein gradient penalty in the adversarial loss to improve the stability of the model during the training phase. Mahapatra and al (2019) ([23]) propose progressive generative adversarial networks (P-GANs). They use a multistage model based on SRGAN with a triplet loss function ([29]), to generate a high scale factor image (from 4 to 32 scale factor) while maintaining good quality. In a previous paper ([24]), they propose to use a salient map to generate images of good quality with a 16 scale factor. Finally, Chaudhari and al. (2018) [2] implemented DeepResolve, a 3D very deep convolutional network using residual learning to generate SR image of slice knee HR. Using 3D convolutions helps to provide additional spatial information and so improves super-resolution performance. Several papers also explore the gain of 3D convolutional network [28], [35].

3 Theory

3.1 Generative Adversarial Network

Introduced by Goodfellow and al. (2014) [8], the Generative Adversarial Network (GAN) is a generative model that follows an adversarial process. The model is composed of two networks, a generative model and a discriminative model. To proceed, let $p_{data}(x)$ denote the distribution of training examples that we want to learn to generate. The generative model G takes as input a random noise $p_z(\mathbf{z})$ and learns a distribution p_g with the mapping function $G(\mathbf{z}; \theta_g)$ where θ_g are the model's parameters. The discriminative model D learns to output the probability p_d that the input $\mathbf{x}$ belongs to p_{data} rather than p_g. Both models are trained at the same time. More formally, D is trained to maximize the likelihood of predicting the correct label to both training examples and samples from G:

$$L_D^{GAN} = \max_D \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x \sim p_g(x)}[\log(1 - D(x))].$$

Whereas, G is trained to maximize the likelihood that G fools D:

$$L_G^{GAN} = \min_G \mathbb{E}_{x \sim p_g(x)}[\log(1 - D(x))].$$

Finally, the model seeks the solution to the following min-max problem:

$$\begin{aligned} L^{GAN} &= \min_G \max_D V(D,G) \\ &= \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] \\ &+ \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))]. \end{aligned}$$

For the back-propagation, we use the block-coordinate descent (BCD): we fix the weights of the first network, and we optimize the other model and vice versa. Since there is no guarantee of balance between the training of G and D, one network may be more powerful than the other (in most cases, D), therefore GAN models are often difficult to train [7].

3.2 SRGAN

SRGAN, a GAN designed for Super-Resolution (SR), has been proposed by Ledig and al (2017) [19]. Below, we describe the architecture of its generator and discriminator.

3.3 Generator

The generator starts with a stack of residual blocks. Each residual block is composed of:

- **A convolution layer:** The kernels of the convolutional layer (the weight matrices), learn to extract features thanks to the convolution operation. We search to extract the most interesting features in the low-resolution image, the features that contain the high frequency details and the structure needed to increase the resolution.

- **A Batch Normalization (BN) layer [14]:** it normalizes layers outputs to reduce the learning time, allows us to use much higher learning rate without having exploding gradients, and reduces the importance of initialization.

  For a given mini-batch, let x be the output of the first layer of a. Firstly, we normalize it:

  $$\hat{x} = \frac{x-\mu_\beta}{\sqrt{\sigma_\beta + \epsilon}},$$

  where $\hat{x}$ is the normalized feature, μ_β the mini-batch mean, σ^2_β the mini-batch variance and ϵ a smoothing term to prevent the division by zero.

  The normalization may change what the layers represent, this is why we add two parameters γ and β to scale and shift the normalized value so that it is optimal:

  $$y = \gamma\hat{x} + \beta,$$

  with y denoting the output of the layer. γ and β are optimized to reduce the loss during the training.

- **A Parametric Rectified Linear Unit (PReLU) layer [11]:** The activation function allows to add non-linearity in the neural network, using a PReLU layer instead of a ReLU layer thus solving the dying ReLU problem where some neurons stay inactive whatever the input is. The PReLU function is define as:

  $$f(y) = \begin{cases} y & y \geq 0 \\ ay & y < 0. \end{cases}$$

  where a is a learnable parameter that controls the negative slope.

- **A convolution layer:** The second convolution layer allows to learn more complex features from the output of the previous layer.

- **A Batch Normalization (BN) layer:** The common layer that follows a convolution layer.

The stack of convolutional layers (with BN layer and PReLU layer for efficient learning) allows to learn and to extract more complex features, essential to a good extraction of all the information in the low-resolution image. Nevertheless, a very deep convolutional network is very sensitive and often has vanishing and exploding gradients which is often solved using residual blocks.

ResNet blocks The ResNet blocks [12] deal with the difficulty of training very deep neural networks and avoid the vanishing gradient

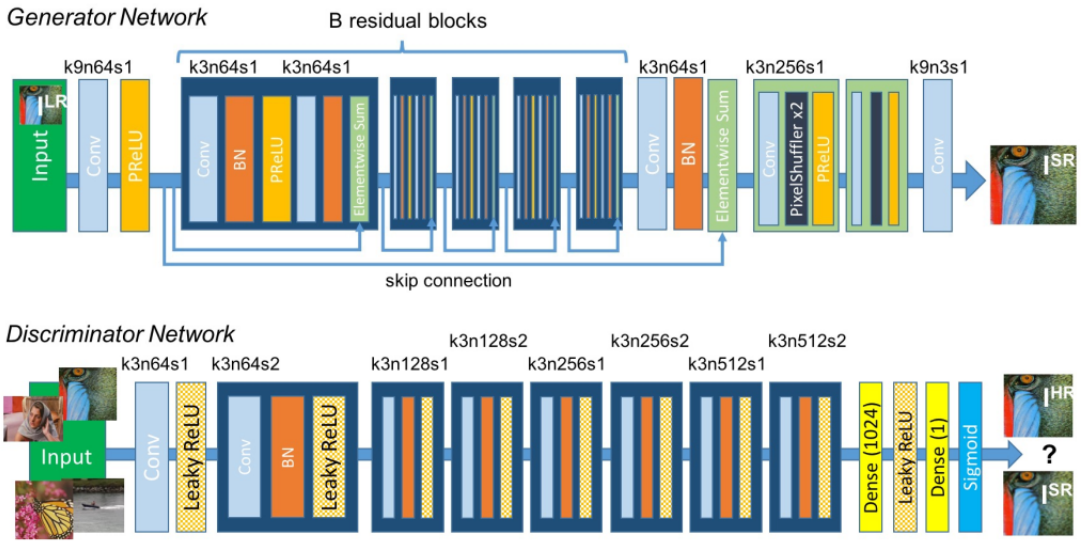


Figure 1: Architecture of Generator and Discriminator Network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer [19].

problem [6]. They use skip connections to learn a residual function $F(x)$ inside of the true output $H(x) = F(x) + x$ (see Fig. 2). Since there is an identity connection x, the layers try to learn the residual $F(x)$. This skip connection allows several important properties [12]:

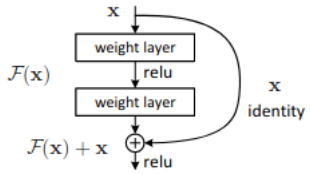


Figure 2: Residual learning: a building block.[12]

- **Simpler learning**: It is easier to optimize the residual mapping than the original mapping. By construction, a deeper model have to produce no higher training error than its shallower counterpart since for a perfectly optimized shallower model, the additional layer has just to learn the identity mapping. With the residual mapping, the learning of the identity becomes easier: it is sufficient to learn a weight matrix equal to a zero matrix.
- **Gradient flow**: For a deep residual network, we have for one ResNet block:

$$x_{l+1} = x_l + F(x_l, w_l),$$

with x_l being the input, x_{l+1} the output and $F(x_l, w_l)$ the residual function. So, for a stack of residual blocks, we have for any deeper block L and any shallower block l:

$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i, w_i).$$

Thus, the chain rule of backpropagation of a ResNet is:

$$\begin{aligned}\frac{\partial L}{\partial x_l} &= \frac{\partial L}{\partial x_L}\frac{\partial x_L}{\partial x_l} \\ &= \frac{\partial L}{\partial x_L}\left(1 + \frac{\partial}{\partial x_l}\sum_{i=1}^{L-1} F(x_i, w_i)\right).\end{aligned}$$

The first term propagates the information directly without passing through the layers whereas the second term propagates information through the layers of the network. Consequently, the gradient of a layer does not vanish even when the weights are small. (see fig. 2)

- **Global residual learning:** Directly connects the input and output images.

Upscaling blocks [31] Once the C feature maps are extracted thanks to the residual blocks, the resolution of these feature maps are increased in the up-scaling block. A up-scaling block is composed of a convolutional

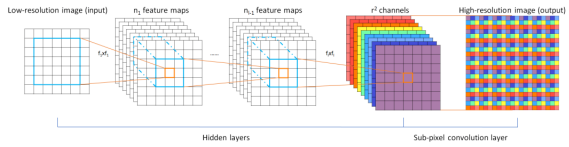


Figure 3: The efficient sub-pixel convolutional neural network (ESPCN), with two convolution layers for feature maps extraction, and a sub-pixel convolution layer that aggregates the feature maps from LR space and builds the SR image in a single step [31].

layer, a PixelShuffler layer then a PReLU layer. The first convolution layer increases the number of feature maps by r^2. Then, the sub-pixel layer [31] rearranges the elements of the $H \times W \times C \cdot r^2$ tensor in a $rH \times rW \times C$ tensor as show in Fig. 3. Finally, the activation layer adds non-linearity.

Each up-scaling block increase the resolution by a factor of 2, so there is $\log(r)$ up-scaling blocks one after the other, where r is the desired factor. The increase of resolution only at the end of the network reduces the computational time since all the previous computations are in the LR space. Finally, a convolutional layer generates the SR image from the C features map of the HR space.

3.4 Discriminator

The discriminator follows the common structure proposed by Radford and al. (2015) [26]: there is a stack of a convolution layers, batch normalization and Leaky ReLU allowing to extract complex feature maps with the number of feature maps gradually increasing from 64 feature maps to 512. Then, the two dense connected layers learn a representation from the feature map to predict if the image is real or fake. The last layer with the sigmoid activation function normalizes the output between 0 and 1 to transform the prediction in probabilities. The discriminator forces the generator to generate an image with the same distribution as the HR distribution thanks to the GAN loss.

Loss function One of the most important improvements of SRGAN [19] is a perceptual loss based on the perceptual similarity computed using the VGG feature maps [32]. The loss of a SR model L^{SR} is commonly based on the mean square error (MSE) loss, with an adversarial loss for the GAN-based methods:

$$L^{SR} = \underbrace{L^{SR}_{MSE}}_{\text{content loss}} + \underbrace{10^{-3} \cdot L^{SR}_{Gen}}_{\text{adversarial loss}},$$

where the pixel-wise MSE loss is calculated as:

$$L^{SR}_{MSE} = \frac{1}{r^2WH} \sum_{x=1}^{rW} \sum_{y=1}^{rH} (I^{HR}_{x,y} - G_{\theta_G}(I^{LR})_{x,y})^2$$

and the adversarial loss is:

$$L^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{theta_D}(G_{\theta_G}(I^{LR})).$$

However, the image generated by the MSE loss often lacks high-frequency content, that leads to overly smooth texture. To solve this problem, the authors propose to replace the content loss by the VGG loss [32] which is closer to perceptual similarity.

$$L^{SR}_{VGG/i,j} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^{SR})_{x,y} - \phi_{i,j}(G_{\theta_G}(I^{LR}))_{x,y})^2$$

with $\phi_{i,j}$ being the j-th convolution layer before the i-th max-pooling layer within the VGG19 network, $W_{i,j}$ and $H_{i,j}$ the respective feature maps dimensions.

3.5 Attention mechanisms

Squeeze-And-Excitation Network -(SE) The output of a convolutional layer contains numerous feature maps (64 to 512 traditionally). They are interdependent, and some of them contain more or less important information for the given task. Despite this, they are all equally used to compute the output of the model. In the considered case of super-resolution, some features map can contain more high-frequency information (basically, the details) essential for the high resolution, but other can contain noise and degrade the final image.

Squeeze and excitation network (SE) [13] is a unit block that allows the network to adaptively re-calibrate the weight of each feature

map. Initially used for the classification task, it has led to a great improvement in results in SR task too. SE blocks contain 3 operations:

- **Squeeze operation:** Global average pooling for each feature map:

$$z_c = F_{sq}(u_c) = \frac{1}{H * W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j).$$

This operation creates a global channel descriptor of the image, where each value is a local descriptor of a feature map.

- **Excitation:** To fully capture channel-wise dependencies:

$$s = F_{ex}(z, W) = \sigma(W_2 \delta(W_1 z)),$$

with σ the sigmoid, δ the ReLU, $W_1 \in \mathbb{R}^{\frac{c}{r} * c}$, $W_2 \in \mathbb{R}^{c * \frac{c}{r}}$ The auto-encoder maps the global descriptor z to a set of feature maps weights.

- **Rescale:** To reweight the feature maps:

$$x_c = F_{scale}(u_c, s_c) = s_c u_c.$$

SE blocks create activation channels adapted to the descriptor z only based on the input: it works like a self-attention function on channels to boost feature discriminability.

3.6 Image Quality Assessment

The quality of an image is difficult to evaluate since it depends on the subjective human perception (eg. how realistic the image looks). Nevertheless, two computational metrics have become established in the field of super-resolution: the Peak Signal-to-Noise Ratio (PSNR), and the Structural SIMilarity Index (SSIM) [36]. The performance of our models has been evaluated on these two metrics.

Peak signal-to-noise ratio - (PSNR) The Peak signal-to-noise ratio (PSNR) was originally used to quantify the reconstruction quality of an image after its compression, hence its natural use for super-resolution. It is defined between two images as:

$$PSNR(I^{SR}, I^{HR}) = 10 \cdot \log_{10} \left(\frac{255^2}{MSE(I^{SR}, I^{HR})} \right),$$

where I^{SR} is the super-resolution image output, I^{HR} is the high-resolution image target and MSE is the Mean Square Error between them [37].

The MSE gives us a measure of the distance between the two images, the ratio allows us to normalize against the maximum pixel value of 255, and so to compare the PSNR of different images (since the signal of an image is rarely normalized). Finally, the log_{10} comes from the large amplitude of the ratio. The PSNR is expressed in dB and when MSE tends to 0, PSNR tends to $infinity$. In practice, it is usually below 40dB. The PSNR is a metric revealing the quality of the reconstruction, it does not take into account the visual perception quality. As a loss, it is preferable to use the MSE rather than the PSNR since the latter is not defined to be positive. Moreover, the logarithm implies a high gradient when the output is far from the target and a flat gradient when the output becomes better.

Structural SIMilarity Index - (SSIM) The second metric was proposed by Wang and al. (2004) ([36]) to measure the similarity between two images based on the structural information. Their assumption is that human perception extracts structural information from a scene (eg. forms, curve, etc.) rather than comparing them pixel by pixel. They define structural changes as variation in image luminance, changes of contrast, etc.

The SSIM index is calculated on various windows of an image. The measure between two windows x and y of size $N \times N$ is:

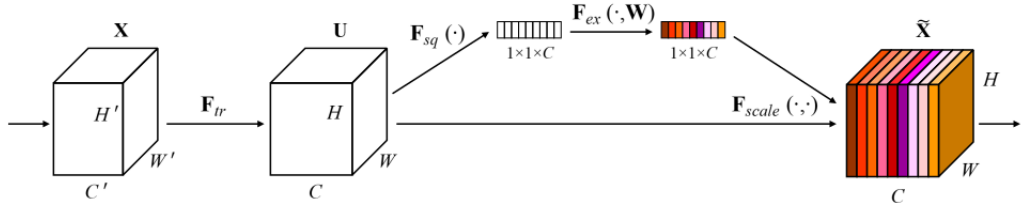
$$SSIM(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma},$$

with $\alpha > 0$, $\beta > 0$, $\gamma > 0$ and c_1, c_2, c_3 are constants to ensure stability when the denominator becomes 0.

- **Luminance:** The luminance of the windows is given by the mean: $\mu_x = \frac{1}{N} \sum_{i=1}^{N} x_i$. So the luminance comparison function is given by:

$$l(x,y) = \frac{\mu_x \mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}.$$

- **Contrast:** The contrast of the windows is given by the standard deviation of the



g. 1. A Squeeze-and-Excitation block.

Figure 4: A Squeeze-and-Excitation block [13].

windows: $\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)}$. So the contrast comparison function is given by:

$$c(x,y) = \frac{\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}.$$

- **Structure:** The structure comparison function is given by:

$$s(x,y) = \frac{cov_{xy} + c_3}{\sigma_x\sigma_y + c_3}.$$

Dosselmann and Yang (2011) ([4]) have demonstrated that the SSIM is directly related to the MSE, which is often unreliable. SSIM compares luminance variation and changes of contrast without taking into account the borders or shapes present. To tackle this problem, SRGAN ([19]) uses a loss computed from the feature maps of the VGG network [32] instead of MSE-based content loss. Since feature maps are the product of convolution functions, they detect shapes and are more invariant to change in pixel space.

It should be noted that these methods are not necessarily consistent with the human visual perception, especially for the GAN-based methods. For this reason, the Mean Opinion Score (MOS) is also commonly used. For this method, human raters give a perceptual quality score to tested images then the average is assigned as the MOS score for each image [37], [19].

Clinical Metrics In the context of medical imaging, visual perception is not relevant. We would like to measure how much the SR bone micro-structure is close to the HR bone micro-structure. So, we can use the clinical trabecular and cortical micro-structural measures and compare them with the true HR image reference. This is the method used by Guha and al (2020) [10].

To compute these measures, we need to reconstitute the 3D volumes on, at least, 110 images to use dedicated biological software. So, the PSNR and SSIM are first used to compare the models for the sake of simplicity, while the clinical metrics will be used later to validate the contribution of super-resolution to medical imaging.

4 Contribution and Results

4.1 Dataset

The dataset used is composed of three different mice bone scans. We used one wild-type mice (WT, phenotype of a species normal form) and two defective mutant mice (Bone Sialoprotein -BSP, and Osteopontin -OPN). The scans were performed with a micro-CT scanner (VivaCT 40, Scanco Medical, Brüttisellen Switzerland) by the SAINBIOSE laboratory. The dataset was acquired at a $10.5\mu m$ and $19.5\mu m$ isotropic voxel size, with 2000 projections and interaction time of 300 ms, at 55KVp (Peak kilovoltage) and 145 μA current.

The scans were performed in the proximal part of the tibia. The proximal area is close to the extremity of the bone and is divided between the epiphyseal area and the metaphy-

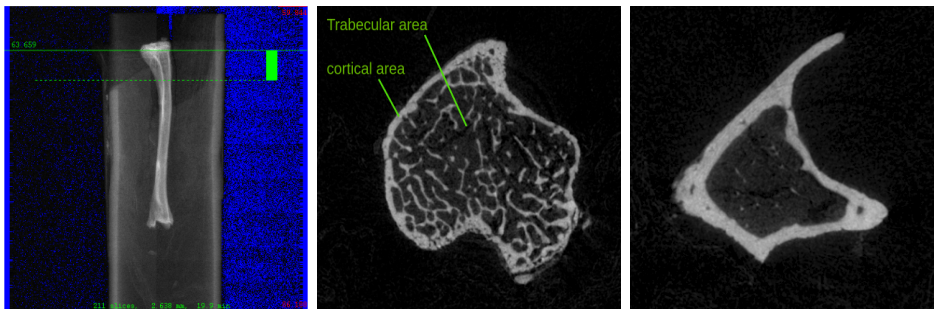


Figure 5: **(Left)** In green, region where the acquisition were performed. **(Middle)** Image HR from the epiphyseal area with numerous trabecular bones and a small cortical bone. **(Right)** Image HR from the metaphyseal area, with few trabecular bones and an important cortical bone.

seal area. So the dataset contains images of both the epiphyseal area, a section composed primarily of trabecular bone (spongy tissue), and the metaphyseal area, a section composed primarily of cortical bone. See Fig. 5.

For each mouse, we have a high-resolution dataset at $19.5\mu m$ and a counter-part low-resolution dataset at $10.5\mu m$, each dataset contains about 200 images. In pre-processing, we performed a rigid registration between the HR and LR images of each mouse. There is no deformation on the LR images since, with a rigid registration, the function is a composition of rotation, translation, and resize by interpolation. The size of all LR and HR images is 500x500 pixel after registration, then we increased the data with rotation of $\frac{pi}{3}$, $\frac{2pi}{3}$ and pi for each image.

4.2 Implementation details

Our SRGAN implementation is inspired by the leftthomas's and eriklindernoren's implementation. The generator contains 16 ResNet blocks, and the output is normalized by using the function $\frac{\tanh(x)+1}{2}$ compared to the original model. For the discriminator, we removed the last block and replaced the sigmoid function by a convolution layer. The loss of the generator is:

$$\begin{aligned} L_G =& L_G^1 + 1e^{-2} * L_G^{GAN} \\ =& ||I^{SR}, I^{HR}||_1 + 1e^{-2} * ||D(I^{SR}), \mathbb{1}||_2, \end{aligned} \tag{1}$$

where $||\cdot||_1$ and $||\cdot||_2$ are respectively the l^1 and the l^2 norm, $\mathbb{1}$ is a one matrix, I^{SR} is the output of G, ie $I^{SR} = G(I^{LR})$, and I^{SR} is the ground truth. $D(I^{SR})$ is a matrix where each element is a probability of truthfulness on a region of I^{SR}.

The loss of the discriminator is (with the same notations):

$$L_D^{GAN} = \frac{||D(I^{HR}), \mathbb{1}||_1 + ||D(I^{SR}), 0||_1}{2}.$$

Training details We trained all our models by using the Adam optimizer [17], with a learning rate $lr = 0.00002$, momentum $b1 = 0.5$ and $b2 = 0.999$ on a 32 batch size during 300 epochs. The implementation is based on PyTorch and each model was trained on the GPU of the Hubert Curien Laboratory cluster. The learning rate was empirically selected after several experiments, to have a fast enough convergence, without divergence of one of the two GAN models. For each model, we select the best model according to the number of epochs on the PSNR score by using cross-validation. The models were trained on random patches of 96×96 pixels for each image of the training set. The training by patch drastically reduces the training time (few hours for 300 epochs against more than 2 days for training on the complete image) and improves the quality of the super-resolution. See fig. 6.

4.3 Experiments on the losses

We did several experiments on the losses of the generator. We didn't use the perceptual loss based on VGG because it has been trained for natural images classification, while we work

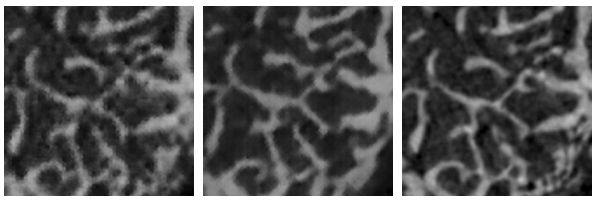


Figure 6: **(Left)** $SR_{\times 4}$ learning by patchs: the image is more noisy, but there is no deformation in the reconstruction For the complete image: (PSNR:**27.367**, SSIM:**0.7521**), mean for the dataset: (PSNR:29.297 $\pm$ 1.499, SSIM: 0.7625 $\pm$ 0.0112). **(Middle)** $SR_{\times 4}$ learning by image. The image is more smoothed, but there are several deformations in the reconstruction. For the complete image: (PSNR:26.045, SSIM:0.6276), mean for the dataset: (PSNR:25.123 $\pm$ 1.382, SSIM: 0.6035 $\pm$ 0.0233). **(Right)** HR reference.

with medical images. We experimented with several coefficients for the adversarial loss and the TV loss. The TV loss is a regularization term that reduces the high difference between neighbor's pixels, so it smoothes the result. PSNR and SSIM results are presented in Table 14. We know that SSIM and PSNR are biased by the MSE and promote smooth results, as confirmed by the outputted images in the Figure 8.

4.4 Squeeze-And-Excitation experimentation

We tried to improve the quality of the reconstruction by using Squeeze-And-Excitation (SE) block. We added a SE block as layer of each residual block. The objective is to re-scale the feature maps to increase the weight of the most interesting of them. Unfortunately, there is no significant improvement as seen in Fig. 9. We experimented with SE on every residual block, and only on the first half of the residual block.

4.5 Learning by intermediate representation for high scale factor

For a 2 or 4 upscale factor, SRGAN can generate SR images close to the ground-thruth HR image. However, for an upscale factor of 8, the reconstruction is severally degraded (see Fig. 10). The generator fails to reconstruct a convincing super-resolution image, and the discriminator detects too easily the difference between a fake generated image and a real HR image, so the training of both models diverges.

The SRGAN generator is essentially a stack of residuals blocks to extract the high-frequency information from the LR image, then in the final steps, there is $log_2(r)$ up-scaling blocks to increase the resolution, with r the upscale factor (see Fig. 1). For small scale factor, it allows to drastically reduce the computation time and it works well since the distance between the LR and the HR image is small. In contrast, for the high upscale factor, the features extracted by the network are probably too far from their HR equivalent. In addition, the feature maps have to be designed for all resolution jumps at the same time since the up-scaling blocks follow each other (3 up-scaling blocks for an 8 scale factor).

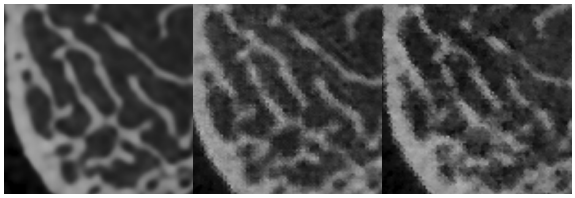
Generator using intermediate representation We propose a generator architecture for a high-scale factor based on a gradual increase in resolution. The generator is composed of an improvement of resolution in 3 steps, where each stage contains a stack of residuals blocks followed by an up-scaling block (see Fig. 11). We think that the scaling by step allows the feature maps of each step to specialize on a single resolution jump. We reduce the learning task into subproblems for which SRGAN is known to be effective, so we reduce the difficulty during adversarial learning. The training by using this generator produces less blurred results, and we can observe that the generator learned to reconstruct the bone micro-structure (see Fig. 12).

However, we can see important hallucination in the SR image. It can be due to intermediate representation, where a small perturbation is propagated in the next representation.

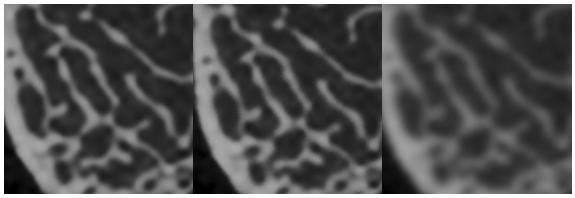
GAN's stack To reduce the error on the intermediate representation, we can try to add control on it. We proposed to use a stack of two GAN's where the first GAN improves the

name exp	L_G	PSNR	SSIM
exp a	$L_G^{GAN} + \mathbf{1e^{-3}} * L_G^1$	32.899 ± 1.368	0.8535 ± 0.0047
exp b	$L_G^{GAN} + \mathbf{1e^{-2}} * L_G^1$	29.297 ± 1.499	0.7626 ± 0.0112
exp c	$L_G^{GAN} + \mathbf{1e^{-1}} * L_G^1$	27.157 ± 1.600	0.5977 ± 0.0056
exp d	$L_G^{GAN} + 1e^{-2} * L_G^1 + \mathbf{1e^{-4}} * L_{TV}$	$\mathbf{32.977 \pm 1.376}$	$\mathbf{0.8693 \pm 0.0063}$
exp e	$L_G^{GAN} + 1e^{-2} * L_G^1 + \mathbf{1e^{-2}} * L_{TV}$	32.256 ± 1.220	0.7774 ± 0.0043
exp f	$L_G^{GAN} + 1e^{-2} * L_G^1 + \mathbf{1e^{-0}} * L_{TV}$	29.075 ± 1.888	0.7729 ± 0.0129

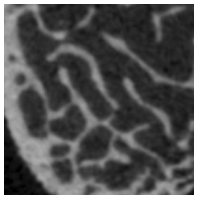
Figure 7: Experiments of several generator's losses.



(a) (**31.326**, 0.8519) (b) (27.622, 0.7535) (c) (25.4288, 0.6096)



(d) (31.344, **0.8661**) (e) (30.873, 0.7863) (f) (26.712, 0.7572)



(g) Ground truth

Figure 8: Visual results for each experiment of Table 7. (PSNR, SSIM) for the complete image, complete images available in appendix, Fig. 18.

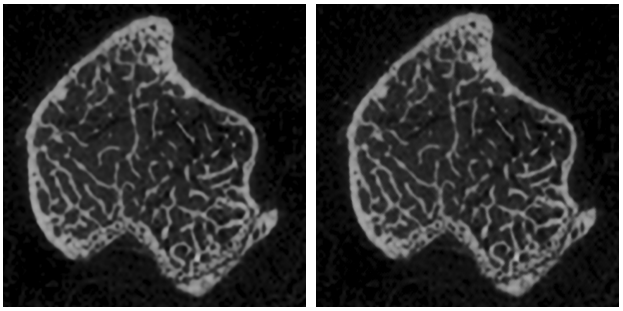


Figure 9: Comparison between a learning with and without SE blocks. There is no significant improvement. Learning during 1500 epochs for the both dataset. **(Left)** Learning with SE block. For the complete image: (PSNR:31.227, SSIM:0.8474), mean for the dataset: (PSNR:32.769±1.341, SSIM: 0.8465± 0.004). **(Middle)** Learning without SE block. For the complete image: (PSNR:31.053, SSIM: 0.8513), mean for the dataset: (PSNR:32.711± 1.401, SSIM: 0.8541 ± 0.0059).

resolution for a factor 2, and the second improves the resolution of a factor 4.

The first generator G_1 generates an intermediate representation, which is used as an input of the second generator G_2. So we have:

$$\begin{aligned} I_{inter}^{SR} &= G_1(I^{LR}) \\ I^{SR} &= G_2(I_{inter}^{SR}) = G_2 \circ G_1(I^{LR}). \end{aligned}$$

The discriminators have the usual losses:

$$\begin{aligned} L_{D_1}^{GAN} &= \frac{||D_1(I_{inter}^{HR}), \mathbb{1}||_1 + ||D_1(I_{inter}^{SR}), 0||_1}{2} \\ L_{D_2}^{GAN} &= \frac{||D_2(I^{HR}), \mathbb{1}||_1 + ||D_1(I^{SR}), 0||_1}{2}, \end{aligned}$$

where I_{inter}^{HR} is an artificial intermediate representation of the I^{HR} obtained by using a rescaling. The second generator G_2 follows the usual loss of SRGAN, with L_G^1 and L_G^{GAN} already defined in Equation 1. For the loss of the first generator G_1, in addition to the usual loss, we add an extra term: $L_{G_2}^{GAN}$, it allows us to propagate the error of I^{SR} through both GAN.

$$\begin{aligned} L_{G_1} &= L_{G_1}^1 + 1e^{-2} * \left(L_{G_1}^{GAN} + L_{G_2}^{GAN}\right) \\ L_{G_2} &= L_{G_2}^1 + 1e^{-2} * L_{G_2}^{GAN}. \end{aligned}$$

A more high-level way to see this term is to say that on the intermediate representation,

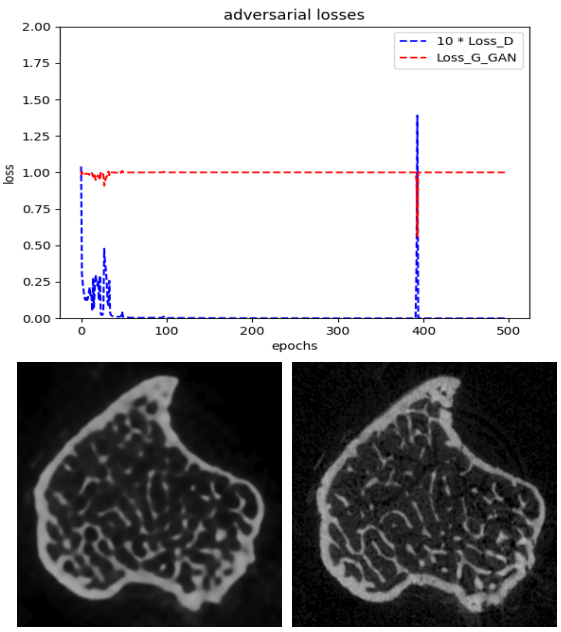


Figure 10: Failure of SRGAN to reconstruct a convincing SR image for a 8 scale factor. **(Top)** adversarial losses of the model. **(Left)** SR image. **(Right)** HR image, ground truth

a small error is weakly taken into account by the losses of the first generator however, it can lead to a large error on I^{SR}. The SR problem is over-parametrized, we re-construct r^2 pixels from 1 pixel, with r the scale factor. So several reconstruction can be produced from the intermediate image, the extra term add constraint on the intermediate representation to reduce the number of bad possible reconstructions.

The results can be seen in the Fig. 13. We note that the obtained result is still far from the true HR and bears a severe deformation. In addition, the second GAN continues to diverge after 1000 epochs (see Fig. 17 in appendix). The PSNR and SISM continue to increase after 1000 epochs due to the first term of the generator loss, the l^1 loss L_G^1.

We also tried with the opposing scale factor for the GANs (4 then 2 scale factor), but didn't notice any improvement in the results. Moreover, it led to an increase of computational time since the size of the image that passes through the second generator is larger.

Discussion Finally, none of these ideas allowed a significant improvement in the quality of the super-resolution images in the case of high-scale factors. The problem seems to come from the low quality of the intermediate representation. There is necessarily a loss of information when we reduce an image by a large factor. So there is probably a limitation in the amount of information that can be retrieved, it a future interesting work to measure the boundaries of information recovery.

Several papers use super-resolution with high scale factor of 8 or 16 [23], [18]. But, their HR images are in high size 1000×1000 or 2000×2000 pixels and their LR keeps the structure and the main information. In our case, our HR images are in 500×500 pixels, with a bone in 200×200 pixels, so the LR lost an important part of the bone micro-structure. In our context, we want to recover high-resolution from a low-resolution image that is severally degraded due to the material limitation. In this regard, it is close to astrophysics problems, where the deep sky objects are coded on only a few pixels and degraded due to the optical limits and we try to reconstruct them.

4.6 Improvement of the clinical results by using SRGAN

Here, we want to study the relationship between resolution and bones micro-structural measures and to calculate the gain that super-resolution brings to the accuracy of bone microstructure measurements. We have chosen different scale factors:

- $\times$**2**: It corresponds (almost) to the actual resolution gap between the LR and HR mice images.
- $\times$**4**: It corresponds to the common super-resolution factor in the literature. A scale factor of 4 corresponds to a $\times 16$ reduction in image pixels. In addition, this will be the resolution gap on Rehan Jhuboo's next working dataset on the astronauts.
- $\times$**8**: It completes the factor set because it allows us to observe what happens in the case of a large resolution gap, which is close to the actual gap between clinical and research imaging.

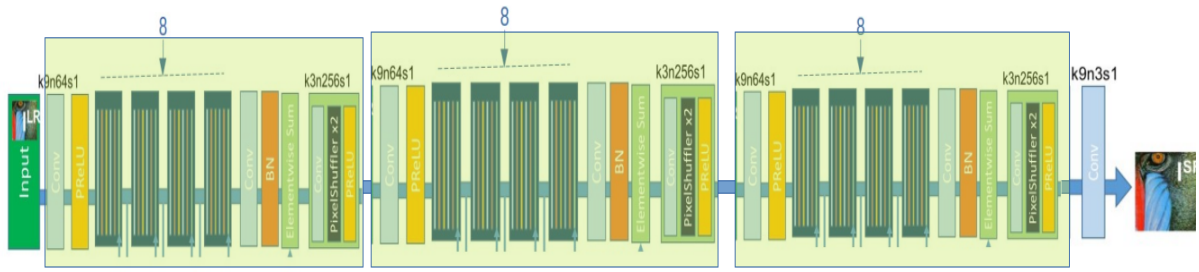


Figure 11: Generator architecture proposed for the high-scale factor. The use of a block containing a stack of residuals block followed by an upscaling block allows generating intermediate representation, which helps to solve the problem of the high scale-factor.

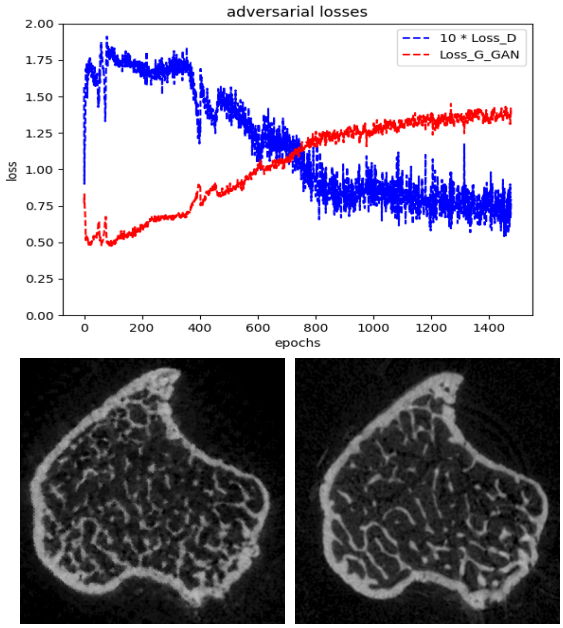


Figure 12: . **(Top)** adversarial loss of the model. **(Left)** SR image. **(Right)** HR image, ground truth

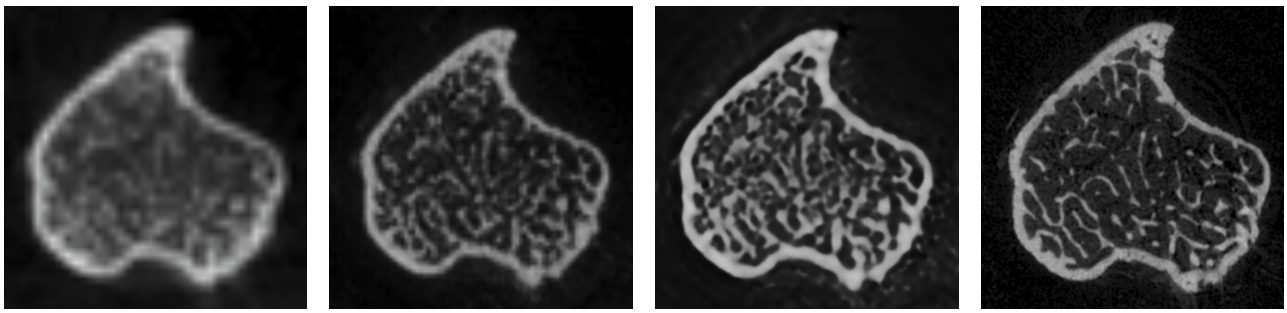
We have only an HR dataset of $10.5\mu m$ and an LR dataset of $19\mu m$. In medical imaging, the function of degradation between HR and LR takes into account noise and unknown degradation due to the limitation of the used scanner. However, in super-resolution benchmarks we usually construct the low-resolution dataset from the high-resolution dataset using interpolation, Gaussian or other degradation method [45], [19], [34]. To this end, and as the ground-truth was not available for all scale factors, we also followed this approach and have generated the LR_{factor} from the direcly $LR_{19\mu m}$ dataset. So the degradation function of LR_{factor} is the scanner degradation of $LR_{19\mu m}$ images plus the degradation of the bi-cubic interpolation.

To investigate our hypothesis that the material degradation function has a strong impact on the quality of super-resolution images and the accuracy of bone micro-structure measurements, we generated an additional LR dataset:

- **$\times 4$ from HR**: a $LR_{\times 4HR}$ dataset generated from the HR dataset, to compare the loss of information and the quality of the super-resolution with the $LR_{\times 4}$ dataset generated from the $LR_{19\mu m}$ dataset.

Training models The HR dataset and each LR dataset are composed of 3 mice, that is 633 images. To compute the micro-structural bone measurements, we need to reconstruct the image of the bone in three dimensions. To respect this constraint, for each LR dataset, we trained a model on 2 mice using data-augmentation during 300 epochs following the implementation details paragraph. Then, we did the selection of the best model according to the number of epochs and the PSNR metric by using one hundred images of the third mice data-augmented dataset as a test dataset. Then, we generated an SR dataset from the third mice without augmentation. We know that a part of the SR dataset was in the test dataset, however, the PSNR metric used during the test time, and the 3D clinical metrics are weakly correlated, this is a small arrangement considering the small amount of data. We used k-fold cross-validation to get for each LR dataset, an SR dataset with 3 mice.

For each dataset, we can see PSNR and SSIM results in Figure 14 and the visual quality in Figure 15).



(a) LR image (b) Intermediate image (c) SR image (d) HR image

Figure 13: GAN's stack results

dataset	PSNR	SSIM
$LR_{\times 2}$	24.891 ± 1.145	0.6136 ± 0.0115
$LR_{\times 4HR}$	$\mathbf{29.504 \pm 1.513}$	$\mathbf{0.7720 \pm 0.0068}$
$LR_{\times 4}$	24.700 ± 1.275	0.5969 ± 0.0140
$LR_{\times 8}$	25.172 ± 1.644	0.6728 ± 0.0287

Figure 14: PSNR and SSIM results for different scale of super resolution. The best result is $LR_{\times 4HR}$ in accordance with its simpler degradation function. The results of the other datasets do not agree with the visual quality of the results (see Fig. 15).

Computation of micro-structural measures We used CT-Analyser (CTAn) Bruker Micro-CT Software. For each mouse, we selected a 3D section in the epiphyseal area for the trabecular analyse and an other 3D section in the metaphyseal area for the cortical analyse (see Fig. 5). We use mask to work only on the trabecular bone in the first case and on the cortical bone in the second case. We did this work for each LR, HR, $SR_{\times 2}$ and $SR_{\times 4}$. All these measures have been computed by Rehan Juhboo. We used linear correlation to measure the improvement of super-resolution the reconstruction of bone micro-structure. The correlation can be see in Figure 16. Unfortunately, the lake of data (3 values for each computation) does not lead to a significant result.

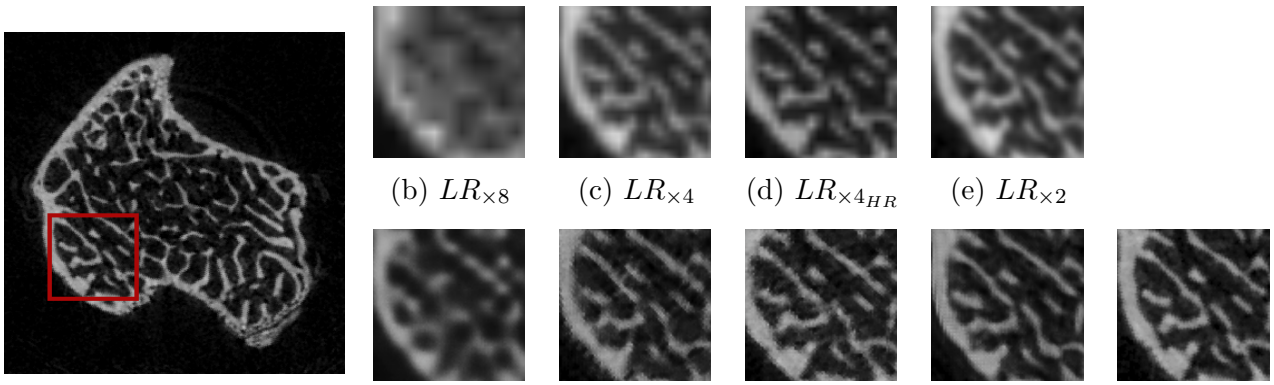
For the $\times 8$ datasets, the image were in too low quality to be used. Since the results of the $\times 2$ and $\times 4$ datasets were not conclusive, it was useless to make the measurements for the $\times 4_{HR}$.

Discussion The method to measure the improvement of super-resolution seems promising but the lake of data is the key issue since we work with only 3 mice. Likely, this problem is easy to solve in the future by increasing our number of mice, then taking several areas of interest in each mouse.

These first analyses have shown that the use of this method on a statistically significant set is quite feasible. In addition, in spite of to the lack of statistical confirmation, the processing of LR images requires additional work time, edges are difficult to detect and measurements are less accurate due to blurring, so this is a first good news for the super-resolution.

5 Conclusion

We have experimented with several ways of improvement of SRGAN. We tried to improve the results by using Squeeze-And-Excitation blocks. Then, we proposed to improve the learning quality in the case of high-scale factors by using intermediate representation. The great difference between the SR natural dataset and our imaging dataset is the low resolution of our HR images. This leads to a critical loss of information in our low-resolution images and a loss of bone structure. This seems to be a strong limitation. Finally, we compute the correlation between bone micro-structural measures to validate the gain of super-resolution for bone micro-structure reconstruction. However, the lack of data makes it impossible to confirm a stronger correlation between SR and HR than between LR and HR.



(a) HR original image (b) $LR_{\times 8}$ (c) $LR_{\times 4}$ (d) $LR_{\times 4_{HR}}$ (e) $LR_{\times 2}$ (f) $SR_{\times 8}$ (g) $SR_{\times 4}$ (h) $SR_{\times 4_{HR}}$ (i) $SR_{\times 2}$ (j) HR

Figure 15: Visual quality results for several factors of scaling. **(Left)** original image. **(Top)** low-resolution input. **(Bottom)** High-resolution counter-part.

×2 **scale**	SR vs TRUE HR		LR vs TRUE HR	
Tb Measures	Linear Correlation (r)	p-value (<)	Linear Correlation (r)	p-value (<)
Trabecular thickness	0.977	0.137	-0.854	0.347
Trabecular separation	0.978	0.134	0.997	0.046
Trabecular number	0.928	0.244	0.966	0.165
Trabecular pattern factor	0.991	0.084	0.905	0.279

×2 **scale**	SR vs TRUE HR		LR vs TRUE HR	
Ct Measures	Linear Correlation (r)	p-value (<)	Linear Correlation (r)	p-value (<)
Tissue volume	-0.800	0.409	0.842	0.362
Bone volume	0.747	0.463	0.181	0.883
Percent bone volume	0.436	0.712	-0.687	0.517
Tissue surface	0.987	0.103	0.985	0.108
Bone surface	-0.231	0.851	-0.532	0.642
Intersection surface	0.837	0.369	0.543	0.635
Bone surface / volume ratio	-0.355	0.769	-0.983	0.115
Bone surface density	-0.954	0.193	0.801	0.408

×4 **scale**	SR vs TRUE HR		LR vs TRUE HR	
Tb Measures	Linear Correlation (r)	p-value (<)	Linear Correlation (r)	p-value (<)
Trabecular thickness	0.940	0.222	-0.455	0.699
Trabecular separation	0.955	0.192	0.955	0.192
Trabecular number	0.935	0.231	0.511	0.659
Trabecular pattern factor	0.945	0.207	0.354	0.770

×4 **scale**	SR vs TRUE HR		LR vs TRUE HR	
Ct Measures	Linear Correlation (r)	p-value (<)	Linear Correlation (r)	p-value (<)
Tissue volume	0.469	0.689	0.427	0.719
Bone volume	0.741	0.468	-0.884	0.309
Percent bone volume	0.892	0.298	-0.838	0.366
Tissue surface	0.990	0.089	0.999	0.023
Bone surface	-0.463	0.694	-0.927	0.243
Intersection surface	0.977	0.137	-0.556	0.625
Bone surface / volume ratio	0.257	0.834	-0.885	0.308
Bone surface density	-0.101	0.936	-0.088	0.944

Figure 16: Correlation of the Trabecular (Tb) and Cortical (Ct) measures between the SR dataset and the HR dataset and between the LR and the HR dataset. The p-values are too high to conclude the strongest correlation of the LR or SR dataset with the HR dataset.

6 Acknowledgments

I thank Ievgen Redko, Marc Sebban and Rehan Juhboo for their help, the numerous discussions and their supervision throughout this internship. It was a pleasure to work with them, and I will keep an excellent memory of this internship. I also thank Alain Guignandon, Norbert Laroche and all the team of Sainbiose, for their help with their scientific expertise in biology. Finally, I thank Richard Serrano and Thibaud Leteno for supporting me.

References

[1] S. Anwar, S. Khan, and N. Barnes. A deep journey into super-resolution: A survey. *ACM Computing Surveys (CSUR)*, 53(3):1–34, 2020.

[2] A. S. Chaudhari, Z. Fang, F. Kogan, J. Wood, K. J. Stevens, E. K. Gibbons, J. H. Lee, G. E. Gold, and B. A. Hargreaves. Super-resolution musculoskeletal mri using deep learning. *Magnetic resonance in medicine*, 80(5):2139–2154, 2018.

[3] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015.

[4] R. Dosselmann and X. D. Yang. A comprehensive assessment of the structural similarity index. *Signal, Image and Video Processing*, 5(1):81–91, 2011.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):142–158, 2015.

[6] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[7] I. Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[9] Y. Gu, Z. Zeng, H. Chen, J. Wei, Y. Zhang, B. Chen, Y. Li, Y. Qin, Q. Xie, Z. Jiang, et al. Medsrgan: medical images super-resolution using generative adversarial networks. *Multimedia Tools and Applications*, 79:21815–21840, 2020.

[10] I. Guha, S. A. Nadeem, C. You, X. Zhang, S. M. Levy, G. Wang, J. C. Torner, and P. K. Saha. Deep learning based high-resolution reconstruction of trabecular bone microstructures from low-resolution ct scans using gan-circle. In *Medical Imaging 2020: Biomedical Applications in Molecular, Structural, and Functional Imaging*, volume 11317, page 113170U. International Society for Optics and Photonics, 2020.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[14] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[15] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[16] X. Jiang, Y. Xu, P. Wei, and Z. Zhou. Ct image super resolution based on improved srgan. In *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, pages 363–367. IEEE, 2020.

[17] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[18] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. 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.

[19] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. 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.

[20] Y. Li, B. Sixou, and F. Peyrin. A review of the deep learning methods for medical images super resolution problems. *IRBM*, 2020.

[21] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[22] Y. Liu, D. Jin, C. Li, K. F. Janz, T. L. Burns, J. C. Torner, S. M. Levy, and P. K. Saha. A robust algorithm for thickness computation at low resolution and its application to in vivo trabecular bone ct imaging. *IEEE Transactions on Biomedical Engineering*, 61(7):2057–2069, 2014.

[23] D. Mahapatra, B. Bozorgtabar, and R. Garnavi. Image super-resolution using progressive generative adversarial networks for medical image analysis. *Computerized Medical Imaging and Graphics*, 71:30–39, 2019.

[24] D. Mahapatra, B. Bozorgtabar, S. Hewavitharanage, and R. Garnavi. Image super resolution using generative adversarial networks and local saliency maps for retinal image analysis. In *International conference on medical image computing and computer-assisted intervention*, pages 382–390. Springer, 2017.

[25] S. P. Mudunuri and S. Biswas. Low resolution face recognition across variations in pose and illumination. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):1034–1040, 2015.

[26] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[27] P. K. Saha, D. Jin, Y. Liu, G. E. Christensen, and C. Chen. Fuzzy object skeletonization: theory, algorithms, and applications. *IEEE transactions on visualization and computer graphics*, 24(8):2298–2314, 2017.

[28] I. Sánchez and V. Vilaplana. Brain mri super-resolution using 3d generative adversarial networks. *arXiv preprint arXiv:1812.11440*, 2018.

[29] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2015.

[30] F. Shahidi. Breast cancer histopathology image super-resolution using wide-attention gan with improved wasserstein gradient penalty and perceptual loss. *IEEE Access*, 9:32795–32809, 2021.

[31] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. 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, 2016.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[33] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[34] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[35] Y. Wang, Q. Teng, X. He, J. Feng, and T. Zhang. Ct-image of rock samples super resolution using 3d convolutional neural network. *Computers & Geosciences*, 133:104314, 2019.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[37] Z. Wang, J. Chen, and S. C. Hoi. Deep learning for image super-resolution: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[38] Y. Xiao, K. R. Peters, W. C. Fox, J. H. Rees, D. A. Rajderkar, M. M. Arreola, I. Barreto, W. E. Bolch, and R. Fang. Transfer-gan: Multimodal ct image super-resolution via transfer generative adversarial networks. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 195–198. IEEE, 2020.

[39] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.

[40] C. You, G. Li, Y. Zhang, X. Zhang, H. Shan, M. Li, S. Ju, Z. Zhao, Z. Zhang, W. Cong, et al. Ct super-resolution gan constrained by the identical, residual, and cycle learning ensemble (gan-circle). *IEEE transactions on medical imaging*, 39(1):188–203, 2019.

[41] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[42] F. Yu, V. Koltun, and T. Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[43] J. Yu, Y. Fan, J. Yang, N. Xu, Z. Wang, X. Wang, and T. Huang. Wide activation for efficient and accurate image super-resolution. *arXiv preprint arXiv:1808.08718*, 2018.

[44] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. 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.

[45] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010.

[46] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.

[47] Y. Zhu, Z. Zhou, G. Liao, and K. Yuan. Csrgan: Medical image super-resolution using a generative adversarial network. In *2020 IEEE 17th International Symposium on Biomedical Imaging Workshops (ISBI Workshops)*, pages 1–4. IEEE, 2020.

7 Annex

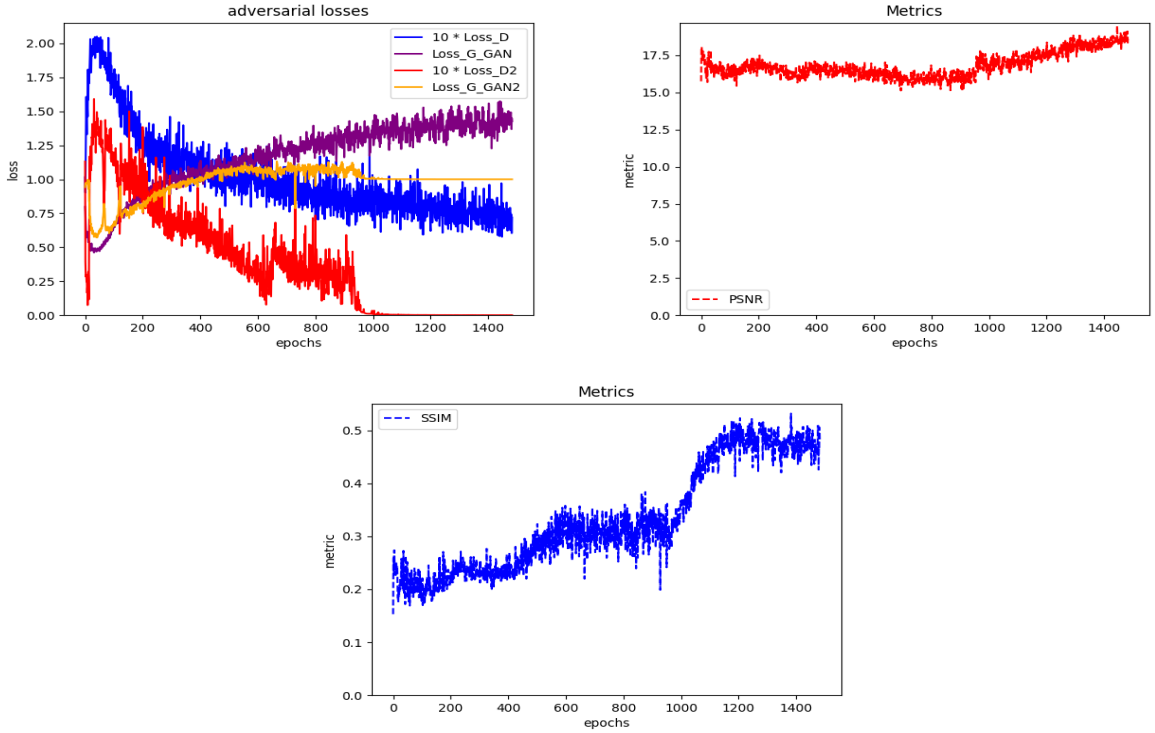
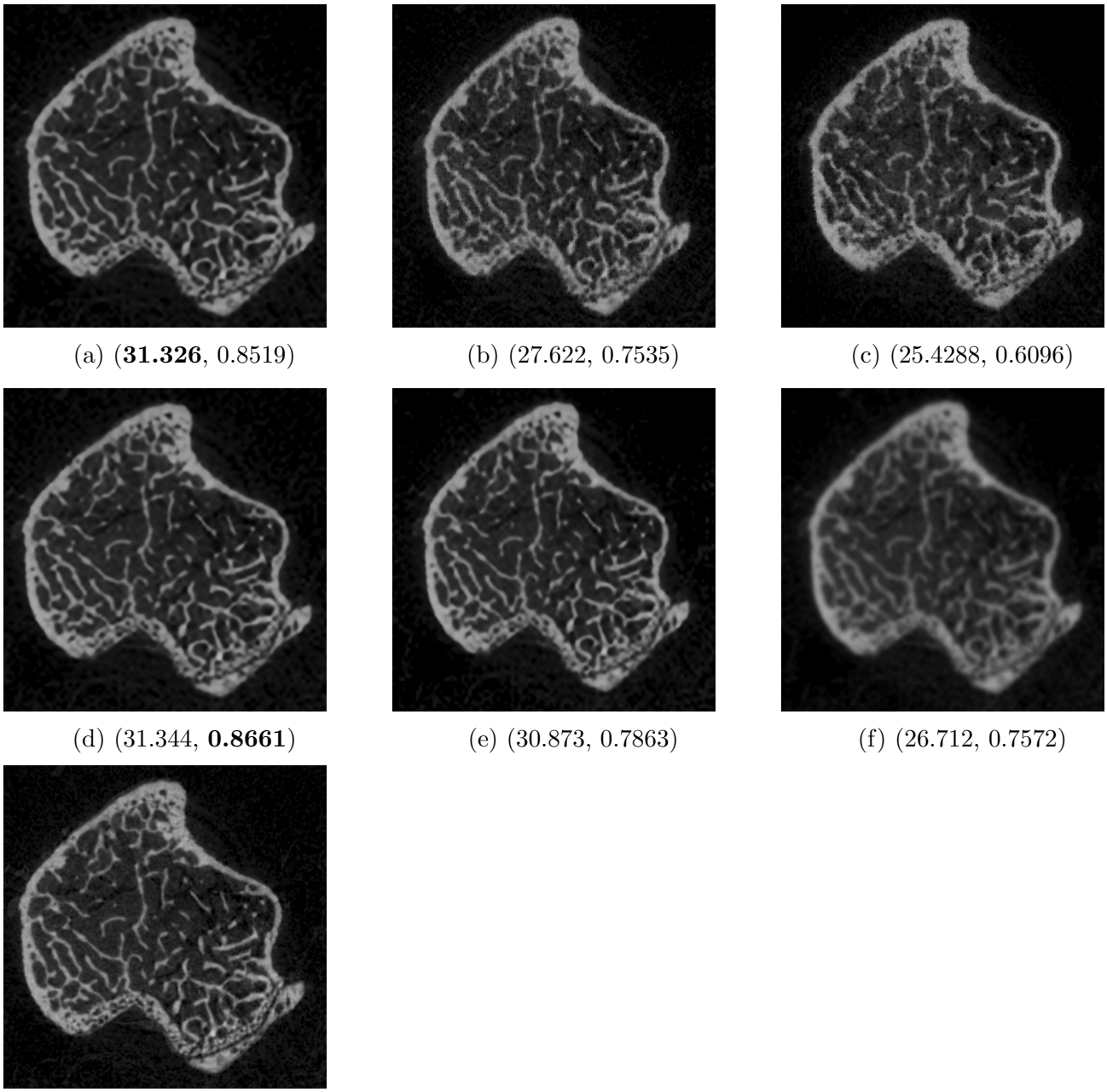


Figure 17: Metrics of learning for the GAN's stack model. **(Top, left)** Adversarial losses. **(Top, right)** PSNR according to the number of epochs. **(Bottom)** SSIM according to the number of epochs.



(a) (**31.326**, 0.8519) (b) (27.622, 0.7535) (c) (25.4288, 0.6096)

(d) (31.344, **0.8661**) (e) (30.873, 0.7863) (f) (26.712, 0.7572)

(g) Ground truth

Figure 18: Losses experimentation. Visual results for each experiments of the table 7. (PSNR, SSIM) for the complete image.