On Using Perceptual Loss within the U-Net Architecture for the Semantic Inpainting of Textile Artefacts with Traditional Motifs

Catalin Stoean*†, Nebojsa Bacanin‡, Ruxandra Stoean*†, Leonard Ionescu §†,
Cristian Alecsa†, Mircea Hotoleanu†, Miguel Atencia¶, Gonzalo Joya¶
*Department of Computer Science, Faculty of Sciences, University of Craiova, Romania
Email: catalin.stoean@inf.ucv.ro, rstoean@inf.ucv.ro
†Artificial Intelligence and Machine Learning Team
Romanian Institute of Science and Technology, Cluj-Napoca, Romania
Email: alecsa@rist.ro, hotoleanu@rist.ro
‡Faculty of Informatics and Computing
Singidunum University, Belgrade, Serbia
Email: nbacanin@singidunum.ac.rs
§Restoration and Conservation Lab
Oltenia Museum, History and Archaeology Section, Craiova, Romania
Email: leo_ionescu@yahoo.com
¶Universidad de Málaga, Spain
Email: {matencia, gjoya}@uma.es

***Abstract*—It is impressive when one gets to see a hundreds or thousands years old artefact exhibited in the museum, whose appearance seems to have been untouched by centuries. Its restoration had been in the hands of a multidisciplinary team of experts and it had undergone a series of complex procedures. To this end, computational approaches that can support in deciding the most visually appropriate inpainting for very degraded historical items would be helpful as a second objective opinion for the restorers. The present paper thus attempts to put forward a U-Net approach with a perceptual loss for the semantic inpainting of traditional Romanian vests. Images taken of pieces from the collection of the Oltenia Museum in Craiova, along with such images with garments from the Internet, have been given to the deep learning model. The resulting numerical error for inpainting the corrupted parts is adequately low, however the visual similarity still has to be improved by considering further possibilities for finer tuning.**

***Keywords*—semantic inpainting; deep learning; cultural heritage; textile artefacts; traditional motifs**

I. Introduction

The moment an artefact is discovered by archaeologists, it undergoes a series of complex restoration procedures in order to be exposed and properly conserved in the museum exhibit. Largely, there are three main points in this pipeline: the chemical and structural assessment of the historical object, the inpainting of its corrupted parts and the 3D digital reconstruction. All these procedures can benefit from the assistance of a computational intelligence model, towards increased objectivity and time efficiency [1], [2], [3].

When the first step of chemical analysis is finished and the restorers perceive the material and degradation state of the artefact, they proceed to cleaning it and re-establishing the missing parts: material, drawing, colors. This process is based on the visual and historical awareness of the expert. Therefore, a second pair of artificial intelligence "eyes", trained on a collection of images of historical objects as similar as possible, should be helpful in examining the context and giving its own semantic inpainting.

Once the computational estimation regarding the initial appearance obtained, the restorer can then deem if it is visually appropriate and decide the proper restoration scheme. According to protocols, an object is allowed to be restored if its missing portions are below 50%. In case the threshold is higher, any intervention would inevitably include the imagination of the expert (or now of the computer) and possibly not be faithful to the ground truth of the object. For these frequent cases, a digital twin of the object can be displayed next to the real object. Semantic inpainting and object reconstruction are also imperative for the consistent efforts towards the digitization of cultural heritage (CH) assets of all grand museums.

It has been nevertheless already acknowledged that archaeology exposes the limits of current computer vision techniques. Since artefacts are highly degraded and with large missing visual content, archaeology is a very challenging application area for artificial vision [4]. The current paper appoints deep learning (DL) to specifically visually reconstruct the content of

This work was supported by a grant of the Romanian Ministry of Research and Innovation, CCCDI – UEFISCDI, project number 178PCE/2021, PN-III-P4-ID-PCE-2020-0788, *Object PErception and Reconstruction with deep neural Architectures* (*OPERA*), within PNCDI III. The work is also partially supported by "Plan Propio de la Universidad de Málaga" and "Integrated Engineering Systems group (TIC-125)"

textile artefacts with traditional Romanian motifs. In addition to the difficulties mentioned before concerning archaeological digital reconstruction, CH costumes have a large disparity of colors on very small areas as opposed to the data within the state of the art for semantic inpainting in general, where there are images of objects that are relatively uniform in appearance (faces, buildings), with large zones having little variation. Hence, the study will investigate the degree to which a U-Net architecture with a perceptual loss is able to accomplish the task of reconstructing semantically plausible images of corrupted garments.

The paper is structured as follows. Section II outlines the surrounding DL methods for semantic inpainting. Section III introduces the creation of the image data set and the description of the type of vest pieces that are found as museum exhibits and constitute the application topic of the paper. Section IV explains the approach undertaken to preprocess the images before DL modelling. The methodology used in the current study is described in section V. The experiments are found in section VI, depicting the entire range, from pre-experimental findings, to the formulation of the projected tasks and considered experimental setup, and ending with the results and their discussion. The conclusions of tailoring a first DL semantic inpainting model for traditional CH items, as well as future steps envisaged to augmenting the current results, are given in section VII.

II. State of the art

Image inpainting assumes two possible tasks. A first one is to remove unwanted parts (scratches, stains, superimposed text), i.e. the image is only locally corrupted. The second target is to synthesize missing parts such that the complete image is plausible both visually and semantically, and the task becomes even more challenging when large content is missing.

Traditional image inpainting techniques fall into one of two categories: local and non-local [5]. The local ones are mostly for single image inpainting and are based on geometry measurement and similarity of patches. Their limitations arise from the lack of fitting pixels to be borrowed from the areas surrounding the missing region and the absence of repetitive objects to infer the properties of the disappeared ones. Additionally, an arbitrary shape and extension of the corrupted area complicate the task. For non-local approaches, similar patches are searched for in external data sets. Here, as well, large missing regions or the absence of a close match in the reference databases hinder the procedure.

DL, on the other hand, can capture detailed (high-level) content of a large area on the base of the spatial context (surrounding pixels), in what is termed as semantic inpainting [5], [6]. Deep models can thus capture both appearance and semantics. The study [6] was the first to introduce a mask of the missing regions of the image for training a convolutional neural network (CNN)-based autoencoder (AE) on the neighboring contextual information. The mask is updated every layer and partial convolutions are used in [7]. More effective for arbitrarily shaped regions and producing sharper content, the paper [5] employs deep generative adversarial networks (GAN) for the task. These models should be however constrained by the provided damaged image; otherwise, they produce completely unrelated pictures. The generator and the discriminator are trained on uncorrupted images. The encoding of the corrupted image that is closest to the image in the latent space is used to reconstruct it with the generator model. Similarity is defined based on a weighted context loss from the "good" pixel count and a prior loss for penalizing unrealistic output in accordance to the training images. The Wasserstein-1 metric for comparing the distributions of the generated versus the real image is often included in the GAN [8]. Boundary equilibrium GAN is believed to be an improved version of this architecture for semantic image inpainting [9]. To also include feature patches present at farther locations from the corrupted part, a contextual attention module, in parallel with dilated CNN, is added to the GAN [10]. A more computationally efficient version of the contextual attention approach with a parallel decoding network is in [11].

Therefore, for semantic image inpainting, literature entries usually envisage faces, natural images (landscape, buildings) and textures. Particularly for CH conservation, a GAN method was employed for art painting restoration [12]. Also, looking more into the archaeological side, a pre-processing U-Net, followed by two GANs - one for segment completion, the other for color estimation - are used in [13] for the restoration of damaged walls.

To the best of our knowledge, there are thus no papers at the intersection of machine learning and CH area for man-manufactured objects. These possess different symmetry, structure and patterns than the natural images treated in the literature. Although some structure should be also present in man-made and decorated objects, there is still a higher intra-class pattern variation for the objects. Moreover, the content of one item on a very small area can be extremely diverse. Hence, the DL models will have more difficulty in "getting the picture" from the real-world collection of historical artefacts.

As a consequence, the models will have to be trained on a large data set of objects from the current restoration area (textile) and with an emphasis on the thematic element, i.e. herein traditional Romanian motifs. Additionally, in order to include the context of the corrupted objects to be inpainted, the available complete parts of these under study will also have to be included into training.

III. Data acquisition

The training and validation data sets have been obtained based on images containing pieces of traditional clothing (especially vests) from various country regions in Romania. They were gathered mainly from the Oltenia Museum, the History and Archaeology Section in Craiova, but also from the web pages of other museums, folklore festivals, collections and blogs.

One example restored at the Oltenia Museum that is part of the test suite is given in Fig. 1, i.e. a man's vest, made of black baize. Its primary role was to offer protection against cold, but it was also decorative, as part of the traditional costume worn for celebrations in the Romanian village. The decorations are geometric and floral, with curved as well as broken lines, showing symmetry and repetition. The colors belong to a wide range. The chemical degradation due to the

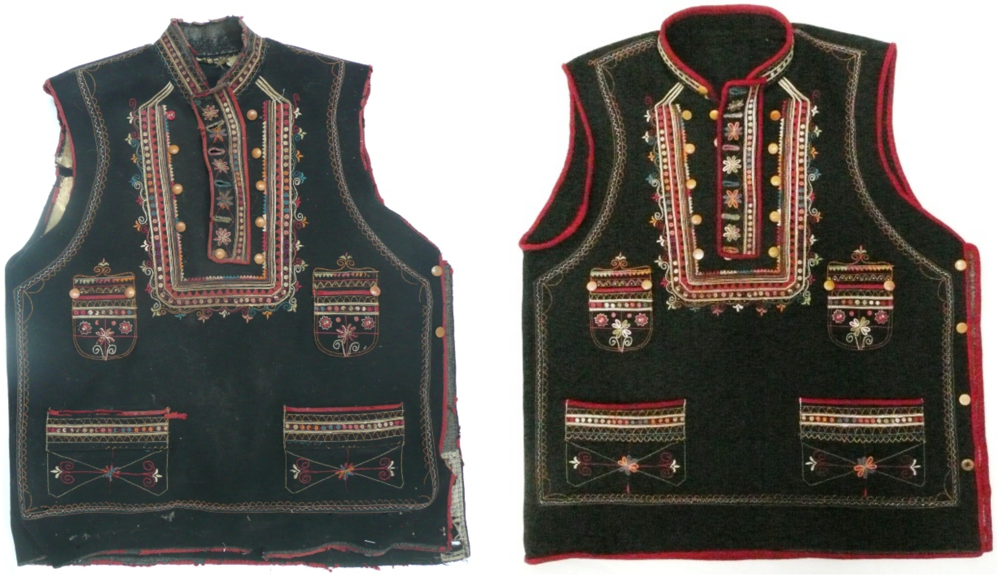

Fig. 1. Example of a man's traditional Romanian vest included in the test of the model, belonging to the collection of the Oltenia Museum in Craiova.

environment led to color fading, while the biological factors (e.g. rodents) generated gap areas.

IV. Data preprocessing

The images have been preprocessed by selecting the rectangular zones denoting only the cloth. Additionally, each image was flipped right/left and top/down and all these versions were kept.

Because the goal of the study was to train an inpainting model, a synthetic data set was generated by randomly cutting out rectangular areas from the raw images. The process was the following:

- Each raw image was resized to a fixed dimension: 256 x 256 pixels.
- For each image, 3 random masks were created by randomly generating up to 5 rectangles (their number is randomly chosen) using the following constraints:
  - The upper-left point of the rectangle is generated between the upper-left corner of the image up to the middle of the image both on width and height.
  - The width and height of the rectangle is generated between 10 pixels and up to half the width and height of the image, respectively.

For the test set, the procedure is similar, with the only difference that there is only one rectangle generated per image, and not up to 5, as it is the case with the training set.

V. Proposed Methodology

For our image inpainting experiments, we consider combining the well-known U-Net architecture with the perceptual loss, due to their versatility on different image transformation tasks.

U-Net [14] is a fast neural network that can easily be trained from end to end by employing very few training image samples. It represents an extension of the convolutional networks introduced by Long et al. in [15]. The architecture contains an encoder that usually consists of a pretrained model (in our case, ResNet34), which passes the input image through convolutional layers and applies the max pooling to learn about the context, and a decoder that is responsible for converting the output of the encoder to the shape of the initial input image. More precisely, the decoder utilizes additional neural upsampling layers which contribute to the propagation through many feature channels of context information, instead of the usual pooling layers. At the same time, the U-Net neural network structure is represented by several multi-channel feature mappings.

For the *perceptual loss* component (also called *feature loss*), we consider the neural network structure related to different objective functions introduced in [16], which is utilized for image transformation tasks. Moreover, it is worth mentioning that the perceptual loss function is based upon the high-level features of a pretrained loss networks. Strictly speaking, the concept of perceptual loss is associated to various loss functions that are actually deep convolutional networks. It should be noted that the underlying loss network is pretrained and it remains fixed during the training process. This network is used to define perceptual loss functions that measure differences in both content and style between input and output images, respectively. The aforementioned loss network consists of the following particular objective functions: a pixel loss function that matches input and output pixels, a feature reconstruction loss which encourages similar representations using pretrained features, and the style reconstruction loss that represents a penalization in the differences in style, color and texture, respectively, for each input-output pair of images, and which is further based on the concept of *Gram matrix*. It must be emphasized that all of these objective mappings are considered with respect to a *base loss* which is represented by the L_1

loss or the *MSE* loss, respectively. That is, given an input-output pair of pixels, for each - the pixel loss, the feature reconstruction loss and the style reconstruction loss - the L_1 loss or the *MSE* loss shall be used correspondingly for the difference between: the pixels, the pretrained features and the values of the Gram matrices associated to pretrained features, respectively.

VI. Experimental Results

The aim of the experiments is to search for adequate values of the parameters controlling the procedure, for determining a good balance between the running time and the accuracy of results. Additionally, we will examine if a small training data set but with image samples containing precise details of the traditional motifs is more appropriate than a larger data set that contains, besides the samples from the small set, pictures taken from a distance, with the complete vest, or an even more comprehensive data set that additionally contains images from other vests that are similar in color and motifs. The data sets will be further referred as being the small, medium, and large ones, respectively.

A. Pre-experimental Planning

Initially, we started with a substantially larger data set holding images from textile artefacts with traditional motifs, i.e. 7938 images in total, but containing many types of clothes, not only vests. These samples were obtained from 200 initial images related to pieces of traditional clothing from various country regions. For these, preprocessing was applied by selecting the rectangular garment area, flips were also applied, similarly to the way they were made for the data set that is used in this study. It is not only that the models took a lot more to train, but we noticed that the results were not visually better than those obtained when the data sets were smaller, but concentrated on a specific type of clothes. Thus, we decided to reduce the amount of data to only a certain type of textiles, i.e. traditional vests, and try the three different scenarios mentioned above.

B. Tasks

We are interested in achieving accurate results in a reduced amount of running time. For this, we intend to examine the output when different quantities and types of samples are used, as well as the performance if the training process is stopped earlier. Additionally, two types of loss are evaluated, the $L1$ and the MSE.

C. Experimental Setup

Our results are based upon *fastai* implementations[1], along with the usage of the flexible *Dynamic U-Net* from *fastai*[2], which utilizes a U-Net neural network that uses state-of-the-art techniques like self-attention layers, along with self-regularized Mish activation functions. Beside these, the *Dynamic U-Net* contains various options for activation functions and normalization layers (in our case we tried *Weight Normalization* and also *Spectral Normalization*), in addition to different settings for avoiding checkerboard artifacts.

For our actual implementations, the masks representing the missing visual content for the image inpainting structure were up to 5 rectangles that can each be in width and height as small as 10 pixels and as large as 50% of the considered image sizes. In addition, for the base loss, we have considered both the L_1 loss and the quadratic *MSE* loss.

For the training process, we have considered running for 50 epochs on images with size 128×128 and batch-size 32, with most of the layers being frozen, except the batch normalization ones. On the other hand, we have unfrozen the layers and then continued training on another 50 epochs. After that, we have doubled the image size and set the batch-size to a lower value, namely 8. For this, we have frozen again all layers, except the batch normalization ones and trained for 50 epochs, followed by the unfreezing of the layers and training for another 50 epochs. A model is saved for each such stage to verify if improvements are indeed reached and the additional training is worth the computational effort.

As for the original U-Net architecture [14], we have considered using different data augmentations for the training image samples. In our case we have added new images based on flipping the training images horizontally and/or vertically. We emphasize that we did not use augmentations based on brightness changes, since these will make the training slower for textile artefacts with traditional motifs, due to the fact that the images are endowed with a complex inherent structure.

The three training data sets contain 228, 144 and 69 samples each. The larger sets include the samples from the smaller ones. The test set is generated from the ground truth images in the 144 data set, but the new rectangles (one per image, in the test set) are randomly generated at different positions. The purpose of generating the test set in this manner was to have a problem as close as possible to the real-world scenario, where a piece of cloth is incomplete, but there are other parts of it that are available and these could be used to assemble a training set. The larger data set simulates a scenario where pictures from other clothes could be used for having samples with similar traditional motifs.

Peak signal-to-noise ratio (PSNR) is finally used to calculate how good are the results from the test set. PSNR represents a quality measurement between the original image and a compressed or a reconstructed image [5]. Higher values for the PSNR correspond to better quality of the reconstructed image. PSNR is computed as shown in (1), where L is the number of maximum possible intensity levels (minimum intensity level is 0) in an image and MSE is the mean squared error.

$$PSNR = 10log_{10}(\frac{(L-1)^2}{MSE}) \qquad (1)$$

D. Results and Visualization

Fig. 2 shows the training sessions for each of the 3 data sets, for the 4 stages (each with 50 epochs) and for both $L1$ and MSE losses.

[1] https://github.com/lgvaz/projects/blob/master/vision/inpainting/lfw_faces.ipynb

[2] https://walkwithfastai.com/Segmentation

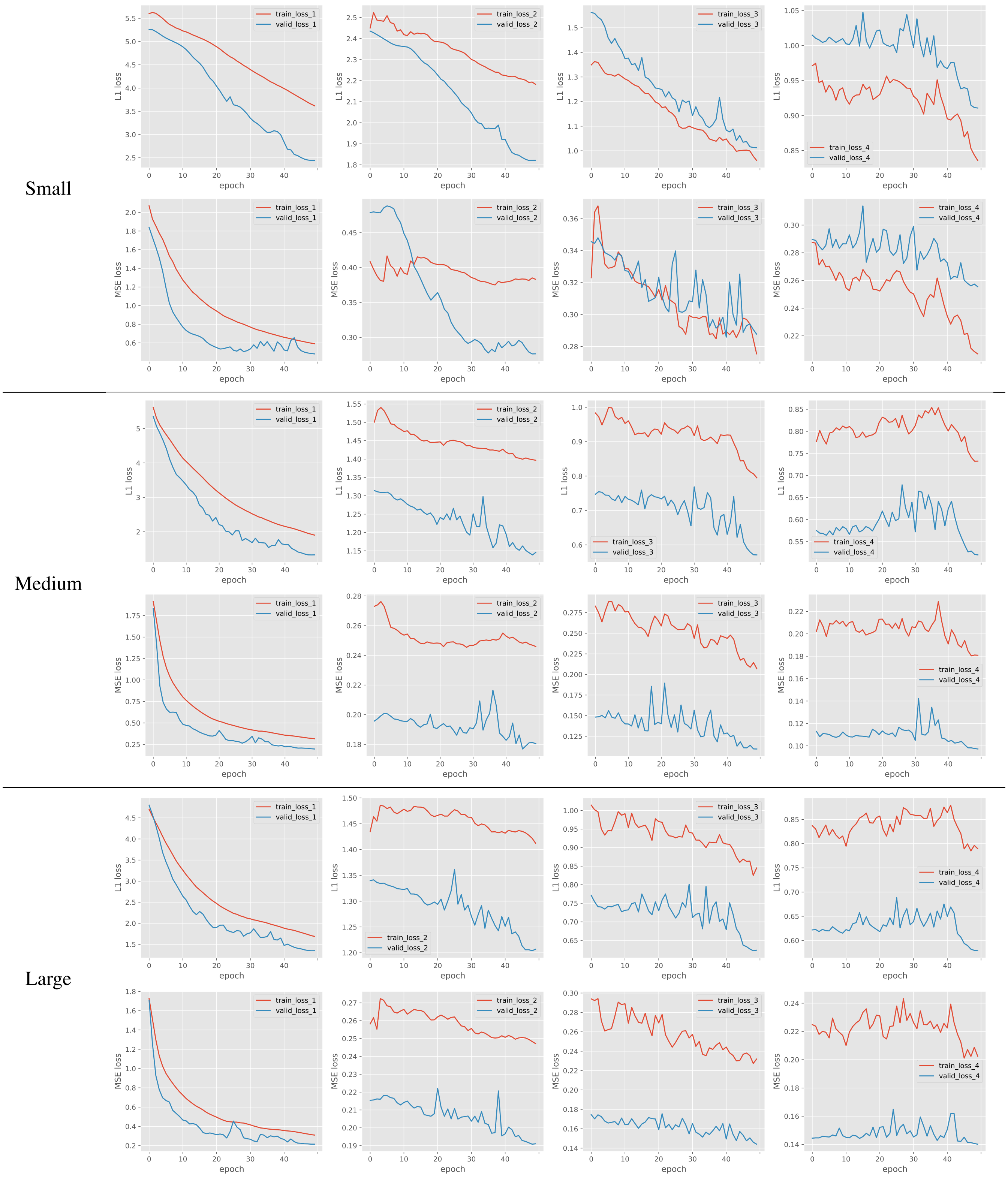

Fig. 2. Training and validation losses for each of the 4 training stages represented on the four columns. The rows alternate the L1 with the MSE losses. First two rows show results on the smaller data set (69 items), next two rows on the middle one (144 samples) and the last two on the largest one (228 images).

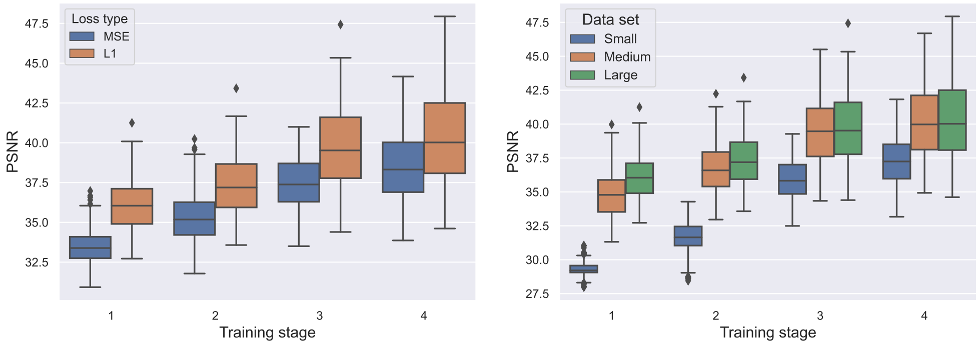

Fig. 3. Box plots showing PSNRs for the larger data set on the left plot when using MSE loss vs L1 loss. The right plot compares the PSNR values when using L1 loss for each of the 3 data sets. Results are shown for each of the 3 training stages. Larger values are better.

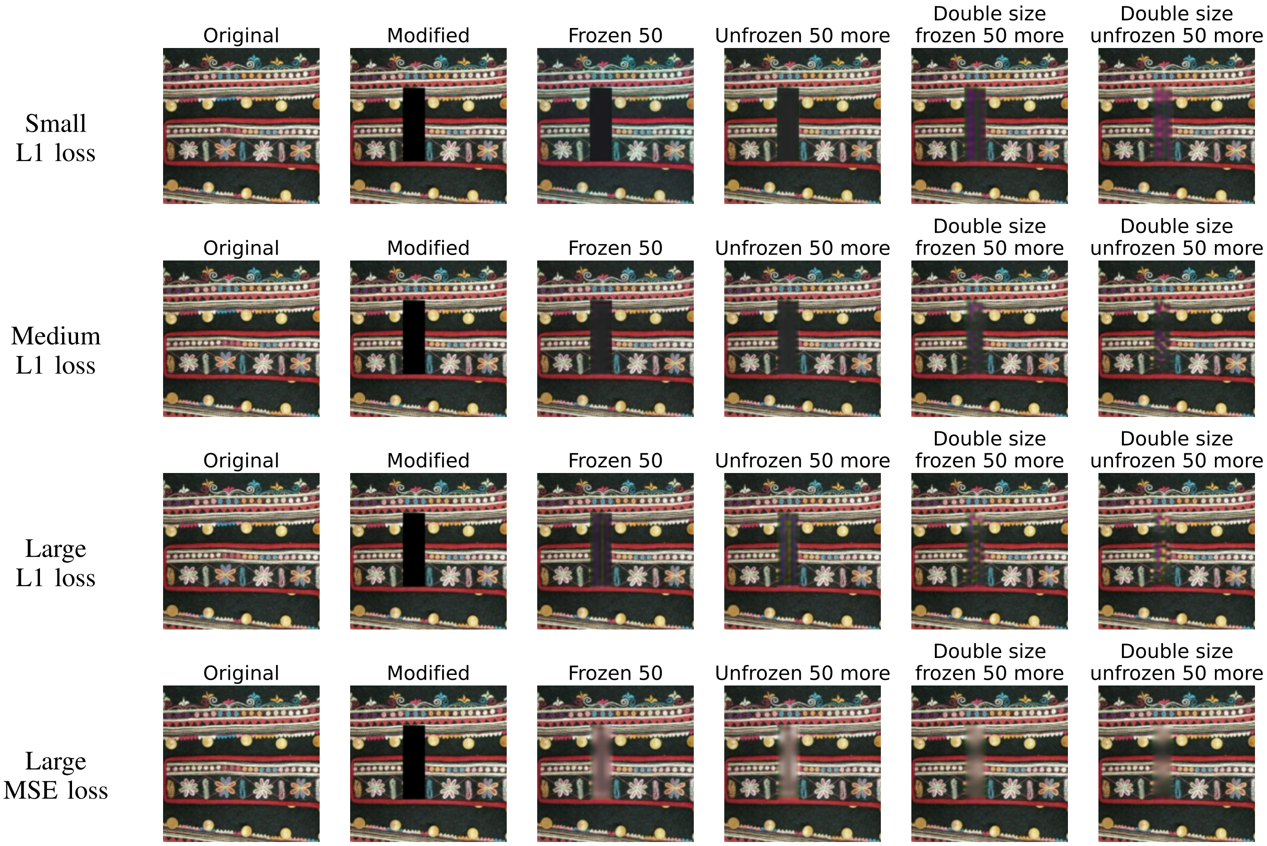

Fig. 4. Results for one sample (left image column) as obtained in different training stages with models that are trained on different data sets. Last two rows show results for the same data set, but using a different loss.

Fig. 3 shows box plots with PSNR results obtained on the test set. The left plot shows the results obtained in the 4 training stages for the larger data set when using MSE and $L1$ as loss. The box plots are obtained by considering the PSNR results for all the images in the test set. The second plot shows the results only for $L1$ loss, since the PSNR results are better when this type of loss is used, for all 3 data sets.

Fig. 4 shows within the first three rows the output obtained for one test sample when $L1$ loss is used for all 3 data sets, while the last row outputs the results when the training is made on the larger data set using MSE loss.

Fig. 5 shows the running time in minutes for each of the three data sets, separated per training stages.

E. Discussion

The results presented in the previous subsection indicate overall that the four training stages indeed lead to the improvement of the accuracy in results. By following from left to right any of the plots in Fig. 2, on the same row, one can see that the decrease in loss continues naturally from left to right. The large differences between the training and validation losses is explained by the fact that for validation we had only

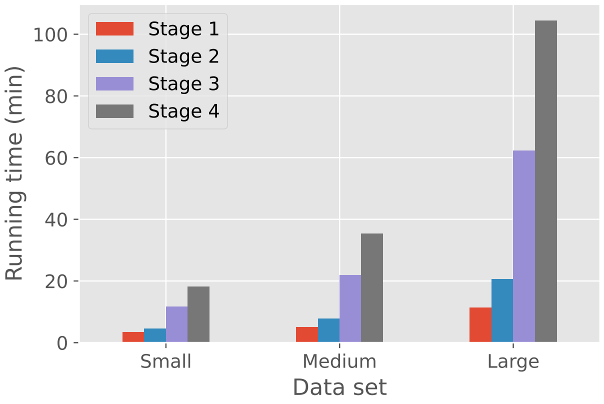

Fig. 5. Running time in minutes for each training stage and for each data set in turn.

one rectangle that hides sections of the images, while for the training there can be up to 5 such rectangles. The loss values should not be compared between MSE and $L1$, since the formulas are different. However, a comparison between the same type of loss can be made between the results obtained when different sizes of data sets are considered for training, since the same test set is used in all cases.

Naturally, the highest decrease in loss is achieved in all cases within the first training stage (first plot column in Fig. 2) and there is no overfitting in any of the cases in these first 50 epochs. By comparing only this initial stage for $L1$ loss (although it happens similarly for MSE) for validation, it can be observed that the larger the data set is, the higher is the decrease in loss. A slight decrease in loss continues in the following 50 epochs (second plot column), as well. Overfitting occurs only in the case the small data set starting with the third training stage, when the images fed into the model are doubled in size. This can be assessed by the change in the training and validation losses, and it becomes even more obvious for the fourth training session. For the medium and large data sets, the training loss remains larger than the validation loss in all cases and there is some small gain in loss in all situations, if we compare the initial loss value from each plot with the final one.

However, for verifying the ability of the model to solve new tasks, Fig. 3 illustrates in the first plot the PSNR for the large data set when using MSE and $L1$ for each of the four stages. The box plots are computed from the individual PSNR values that are obtained for each image in the test set. In all 4 cases, the $L1$ values dominate the ones of the MSE. A T-test for two independent variables is used for comparing the results for each stage in turn and the PSNR values are significantly better when $L1$ is used as compared to the case when MSE loss is considered in all cases. For instance, for the fourth case, when the averages are closer, the p-value is 5.8e-09, which is smaller than the threshold 0.05.

Accordingly, we only use $L1$ for the second plot in Fig. 3, where the results are compared when the training is made on the different data sets. The PSNR result values for the small data set are in all 4 cases smaller than the ones obtained by the model that used the medium data set for training and this is also verified by the statistical test. However, the difference between the model trained on the medium data set and the large one is significant only in the first 2 stages (for the second stage with a p-value of 0.003), for the other two having p-values of 0.49 and 0.65, respectively. The actual mean values for the third stage for the models trained on medium and large sets are 39.49 and 39.69, while for the fourth stages they are 40.11 and 40.25, respectively.

Fig. 4 goes in line with the results exhibited in both plots from Fig. 3. Still, although the PSNR outputs show that, in general, the results between the third and fourth stages are similar, the last pair of columns from Fig. 4 outlines that for this particular case the last training stage led to outcome improvement. When measured via PSNR, the computed value between the original image and the image for row *Large L1 loss* with the title *Double size frozen 50 more* is 38.12, while the PSNR that compares the original and the image from the last column in the same row is 38.6. This indicates that the image from the last column is indeed better, but not by a large degree. For comparison, from the same row, the corresponding PSNR values for the first stage image is 34.79, while for the second stage is 35.84. When computing for this image alone the result obtained after all four training stages by the model trained on the medium data set, the PSNR value was of only 35.84, so it does not reflect well the overall result from the second plot in Fig. 3.

Apart from the numerical values of the metrics, the expert perceptual evaluation is the final performance indicator. Although the millimetric detail is not found precisely, the inpainting is promisingly approaching the desired content. It is clear that the human eye will also take into account symmetry and also sequence repetitions, which the model cannot seize.

As concerns the running time for each training stage and for each data set, Fig. 5 indicates that, when the size of the images are doubled, the running time increases to more than twice the previous one. Also, the larger data sets need significantly more time. Still, the medium data set, the one that reaches the relatively similar results with the larger one after all training phases, does not consume a very large amount of time. In total, for all 4 training stages, the medium data set needs approximately 70 minutes, which is a reduced running time for such a task. The runs are made using an RTX 3090 nVidia GPU.

VII. Conclusions and Future Work

The current study put forward a first approach for a deep semantic inpainting of traditional textile artefacts. Given the multitude of colors and motifs on these garments, as well as the symmetry that has to be visually obeyed, the inpainting task is not as straightforward as is with general objects from the literature.

An U-Net holding a perceptual loss was tailored for the problem and the resulting loss and PSNR values are good, also for a data of only 144 samples to learn from and 70 minutes training time. Nevertheless, when turning to the visual perception of the result, there is definitely a need for further local exploitation concerning drawing and colors.

A first idea would be that the training masks should be smaller, e.g. at a maximum of 10% of the original image size. Considering masks that have irregular shapes rather than only rectangles also represents a scenario that needs to be tested. The next larger step to be tried would be an entirely different approach, such as generative adversarial networks, to the semantic inpainting.

References

[1] C. Stoean, L. Ionescu, R. Stoean, M. Boicea, M. Atencia, and G. Joya, “A convolutional neural network as a proxy for the xrf approximation of the chemical composition of archaeological artefacts in the presence of inter-microscope variability,” in *16th International Work-Conference on Artificial Neural Networks (IWANN), Advances in Computational Intelligence*, 2021, pp. 260–271.

[2] R. Stoean, L. Ionescu, C. Stoean, M. Boicea, M. Atencia, and G. Joya, “A deep learning-based surrogate for the xrf approximation of elemental composition within archaeological artefacts before restoration,” *Procedia Computer Science*, vol. 192, pp. 2002–2011, 2021.

[3] R. Stoean, N. Bacanin, L. Ionescu, M. Boicea, A.-M. Gărău, and C.-C. Ghiţescu, “Semantic segmentation for corrosion detection in archaeological artefacts before restoration,” in *2021 23rd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 2021, pp. 246–251.

[4] N. Derech, A. Tal, and I. Shimshoni, “Solving archaeological puzzles,” *Pattern Recognition*, vol. 119, p. 108065, 2021.

[5] R. Yeh, C. Chen, T. Y. Lim, A. Schwing, M. Hasegawa-Johnson, and M. Do, “Semantic image inpainting with deep generative models,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6882–6890.

[6] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, “Context encoders: Feature learning by inpainting,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2536–2544.

[7] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, “Image inpainting for irregular holes using partial convolutions,” in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 89–105.

[8] P. Vitoria, J. Sintes, and C. Ballester, “Semantic image inpainting through improved wasserstein generative adversarial networks,” in *14th International Conference on Computer Vision Theory and Applications (VISAPP)*, 01 2019, pp. 249–260.

[9] Y. Jia, Y. Xing, C. Peng, C. Jing, C. Shao, and Y. Wang, “Semantic image inpainting with boundary equilibrium gan,” in *Proceedings of the 2nd International Conference on Artificial Intelligence and Pattern Recognition*, ser. AIPR ’19. New York, NY, USA: Association for Computing Machinery, 2019, p. 88–92.

[10] J. Yu, Z. Lin, J. Yang, X. Shen, and X. Lu, “Generative image inpainting with contextual attention,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 06 2018, pp. 5505–5514.

[11] Y.-G. Shin, M.-C. Sagong, Y.-J. Yeo, S.-W. Kim, and S.-J. Ko, “Pepsi++: Fast and lightweight network for image inpainting,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, pp. 1–14, 03 2020.

[12] N. H. Jboor, A. Belhi, A. K. Al-Ali, A. Bouras, and A. Jaoua, “Towards an inpainting framework for visual cultural heritage,” in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2019, pp. 602–607.

[13] Y. Ibrahim, B. Nagy, and C. Benedek, “Deep learning-based masonry wall image analysis,” *Remote Sensing*, vol. 12, no. 23, 2020.

[14] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[15] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[16] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *European conference on computer vision*. Springer, 2016, pp. 694–711.