Deep learning approach for artefacts correction on photographic films

Strubel David[1], Blanchon Marc[1] and Fofi David[1]
[1]ImViA Laboratory, ERL VIBOT CNRS 6000, Université de Bourgogne Franche-Comté (UBFC)

ABSTRACT

The use of photographic films is not totally obsolete, photographers continue to use this technology for quality in terms of aesthetic rendering. A crucial step with films is the digitization step. During the scanning process, dust, scratch and hair (artefacts) are a real problem and greatly affect the quality of final images. The artefacts correction has become a challenge in order to preserve the quality of these photos. In this article, we present a new method based on deep learning with an encoder-decoder architecture to detect and eliminate artefacts. In addition, a dataset has been created to carry out the experiments.

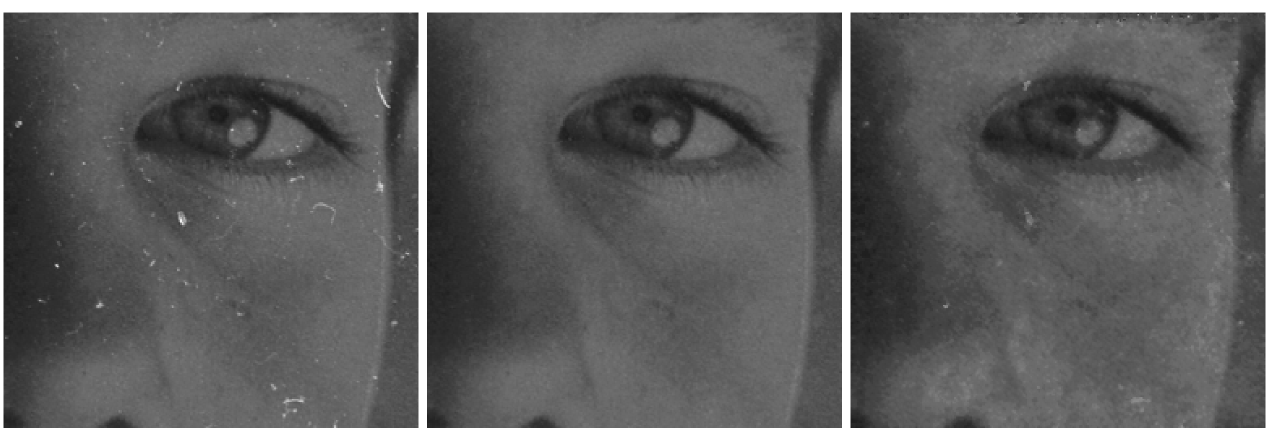
Keywords: artefact removal, photographic film, deep learning, quality control

1. INTRODUCTION

Nowadays photographers and film-makers continue to use analogue films for different reasons, as image quality or out of nostalgia. One of the most crucial steps in the use of analogue film is digitization. Indeed, to exploit the full potential of films, it is necessary to digitize images in high quality in order to use modern post-production tools. When scanning films, scratches, dusts and/or hairs can affect the quality of final images. The scanning process is also long and expensive. The standard artefact reduction approaches such as clean-room development or the correction on film are extremely expensive and tedious tasks. In order to overcome this problem, a solution using a convolutional neural network is proposed to restore the images. In this article, we present our dataset[7] and the preliminary results obtained using a SegNet[8] deep learning network.

2. STATE OF THE ART

Removing artefacts on films has long been a challenge. One of the solutions proposed so far has the disadvantage of dividing the problem in two parts. Firstly, the segmentation of artefact and on the other hand, in-painting. Richard et al.[6] proposed an efficient painting method based on manual selection of imperfections. This method



(a) Input (b) Ground Truth (c) Prediction

Figure 1: A (c) corrected image obtained for a consequent quantity of artefacts on the (a) input image and comparison with the (c) ground truth.

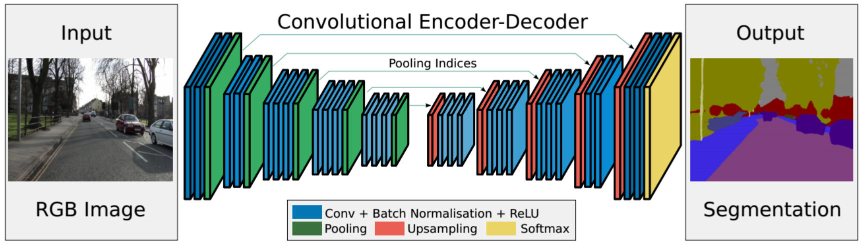


Figure 2: Illustration of SegNet[8] architecture.

then proceeds to convolutions for in-painting. More recently, Bergnan et al.[2] proposed a fully automatic solution to detect dust, scratches and hair and then remove those artefacts. In the work of Bergnan et al.,[2] artefacts detection is done locally to provide good quality segmentation of imperfections by pixel labelling method. Besserer et al.[3] also offers film correction by focusing on the vertical stripes (scratches) commonly observed on photographic or video films. Due to a mechanical defect in the cameras, the films may be scratched several times but always vertically. The solution provided is based on vertical scraping in order to easily detect and correct the artefacts physically generated. A similar problem of artefacts on films is the digital sensors dusts: Zhou et al.[4] or Dirik et al.[5] proposed a solution to overcome this specific problem. The more interesting part in these articles is the segmentation of the sensors dusts which is close to the segmentation of dusts and hairs on films.

3. PROPOSED DATASET

The dataset consists of several images scanned with appropriated sensors, mainly the Epson v750, which allows a high resolution of 6400dpi (about 12000px wide for a total of approximately 150 megapixels on 6×6cm films). The images are scanned to be digitized which induces several dust, hair and other artefacts. Due to this size, the images cannot be used directly for training. The solution adopted consists in splitting and cropping the images in 256px square sub-images. The Ground Truth (GT) is handmade by a photographer which have to aim at the manual removal of all imperfections. The dataset used is available on Github[7] . In this dataset, all images are converted to grayscale. The final dataset consists of 2,708 images with 2/3 for training and the last 1/3 for randomly distributed validation.

4. SEGNET

The SegNet is a deep fully convolutional network based on encoder-decoder architecture for image segmentation; It was primarily designed for scene understanding. The encoder is based on the VGG16 topology[8] apart from the fully connected layers. The VGG16 is a well known architecture for object classification and the usage of a pre-trained model for generalized object classification reduce greatly the training time. The encoder is composed of several layers (13 layers). The encoder layers are composed of convolutions with batch normalization and rectified by ReLU (see SegNet network topology in Figure 2). After few convolutions, normalization and rectification max pooling is made to sub-sample the input. The max pooling help the network to be more generalized. Once this set of convolutions and max pooling is performed (see network topology in Figure 2), the encoder step is achieved. The goal of the decoder is to convert the object classification made thanks to the encoder and turn it into pixel-wise segmentation. Due to that, the input of the decoder has to be up-sampled to return to the original size of input image. To convert classified features to pixel-wise segmentation, a set of convolutions and up-sampling is performed several time. This allows converting space from features to pixel labels. The size is recovered through the step of up-sampling using encoder indices previously saved. To process the up-sampling, the indices of the max-pooling have been memorized to subsequently be used for the unpooling. The indices memorization allow to have better boundary of the segmented regions during the up-sampling. The SegNet is an efficient convolutional encoder-decoder architecture for image segmentation and requires a reasonable training time thanks to the initialisation based on VGG16.

From the previously defined network, it can be deduced that segmentation can be well performed using SegNet architecture and can accommodate the requirement of our task. We propose to use a simple implementation of the network in PyTorch. The results shown are based on a fast training of the network of around 150 epoch using 256 classes. The number of classes correspond to the possible intensity of the pixel (between 0 and 255) plus a class of overflow required for the training. In order to optimize the network weights, an Adam[10] is used with a low learning rate at 10^{-4}. The overall loss for the network will be a cross entropy loss.

5. RESULT AND TALK

Our hypothesis is to claim the potential effectiveness of convolution networks for the artefacts correction. The method is effective for the two main problems which are the segmentation of artefacts and the in-painting procedure. A SegNet has been used because of its segmentation capabilities and its simplicity of implementation. In fact the segmentation of the artefacts is the more complex subtask and then in-painting is performed by the same network according to the detected defects. Consequently, a quick training (only 100 epochs) was done to evaluate our hypothesis. The result is proposed on 50 images that are not used for training. To evaluate the quality of the results, a MSE (Mean Square Error) metric was used (see the affiliated distribution in Figure 3) as well as standard metrics commonly used in deep learning such as mIoU, F1 score, accuracy and recall (see Table 1 and Figure 5). As shown in Figure 6, visual results tends to shown appropriate prediction and the majority of artefacts are removed or attenuated.

The average MSE recovered from the test-set is 338.92. The image presented in Figure 1 is then representative of the average with 244.62 imperfections despite a high number of artefacts. The minimum imperfection rate of the test-set is 643.51 and the max is 39.92.

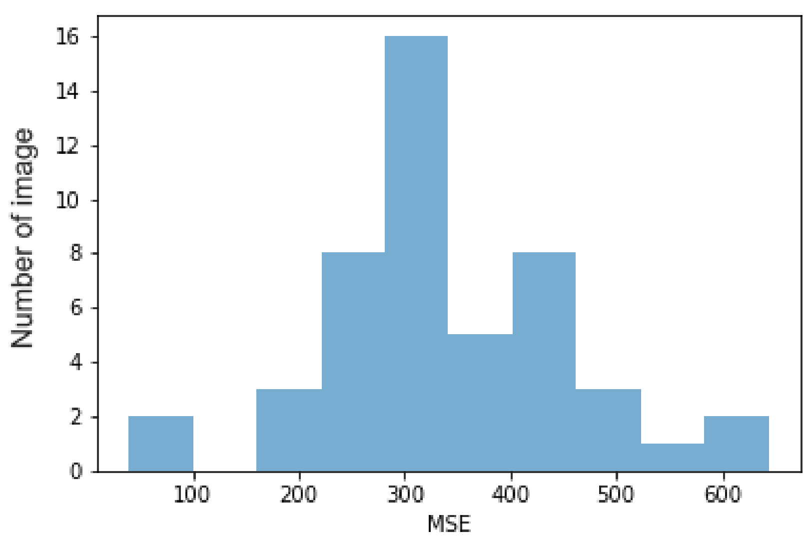
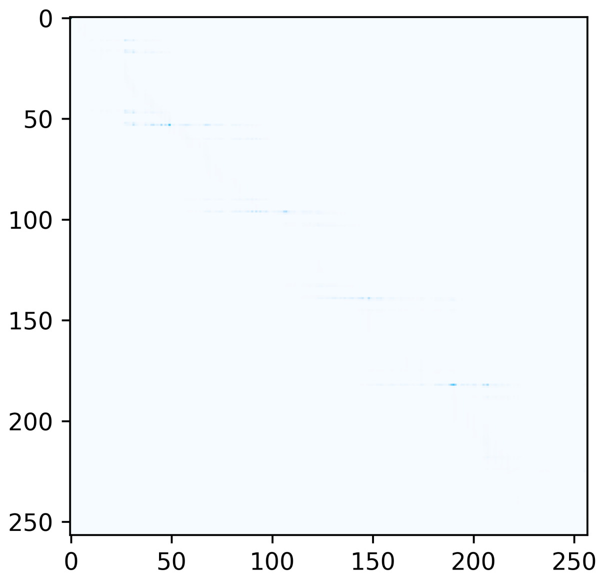
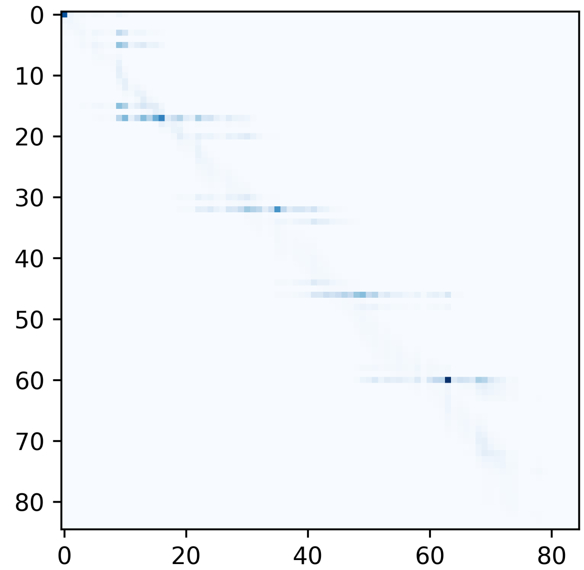


Figure 3: Distribution of MSE for the test set.

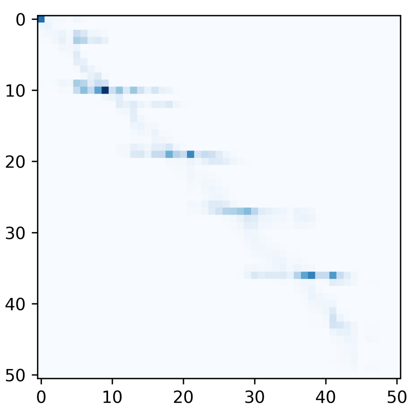
To go further, the quality of the images correction is evaluated with using IoU, F1 score, recall and precision. These standard metrics have been computed on the test set. All these indices are based on the confusion matrix. The confusion matrices is used to evaluate the classification quality (see in Figure 4.a). The results obtained for these metrics (IoU, F1 score, recall and precision, see Table 1), for the 256 classes appear not really efficient (see the figure 5 for 256 in horizontal axis). The weak efficiency is due to the segmentation for 256 classes which is a high number of label. A strategy is to merge close values to reconsider the number of classes. It is induced by the high semantic relationship between adjacent classes. The class reduction is the sum of line and columns in the



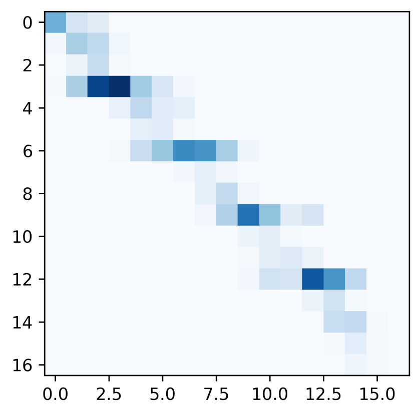
(a) 256 class



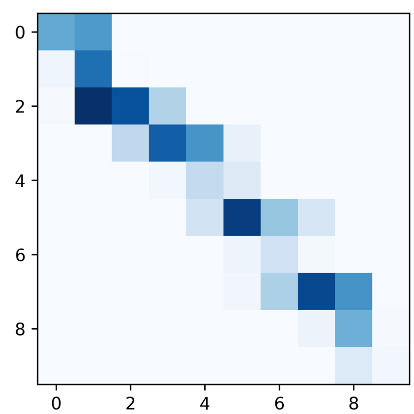
(b) 85 class



(c) 50 class



(d) 17 class



(e) 10 class

Figure 4: Confusion matrices for the test set. The confusion matrix is computed for different number of class by fusion of adjacent classes.

confusion matrix with a desired deviation. It is similar to a convolution with a stride value equal to deviation. This leads to the creation of bigger class which are directly proportional to the chosen neighbourhood. The idea here is to embed multiple values and therefore reduce unnecessary precision and error propagation. This could be considered as optimising a segmentation with less classes. Indeed it is possible to affiliate this process as considering one class and its neighbours as the exact same value. The confusion matrices are displayed in the Figure 4. In Figure 4.a the original confusion matrix followed by confusion matrix with fewer and fewer classes in Figure 4.b to 4.e. Consequently the agglomeration of the neighbourhood and the resulting reduction classes highlight a much better efficiency.

6. FUTURE WORK BASED ON IN-PAINTING

Thanks to the presented work, the deep learning appear suitable and promising for the problem of artefact removal. The solution proposed is based on the segmentation of artefacts. In fact, as already discussed previously, detection, followed by artefacts segmentation, is the most complex task. This has led to the usage of a

MSE	$\frac{1}{n}\sum_{i=1}^{n}(x-\hat{x})^2$
F1	$\frac{2TP}{2TP+FP+FN}$
Precission	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TN+FN}$
IoU	$\frac{TP}{TP+FP+FN}$

Table 1: The metrics commonly used are defined based on the True Positive(TP), False Positive (FP), False Negative (FN), True Negative (TN), computed form the Confusion matrix form the Figure 4.
The Mean Square Error (MSE) is estimate where x is the GT images and $\hat{x}$ is the predicted images.

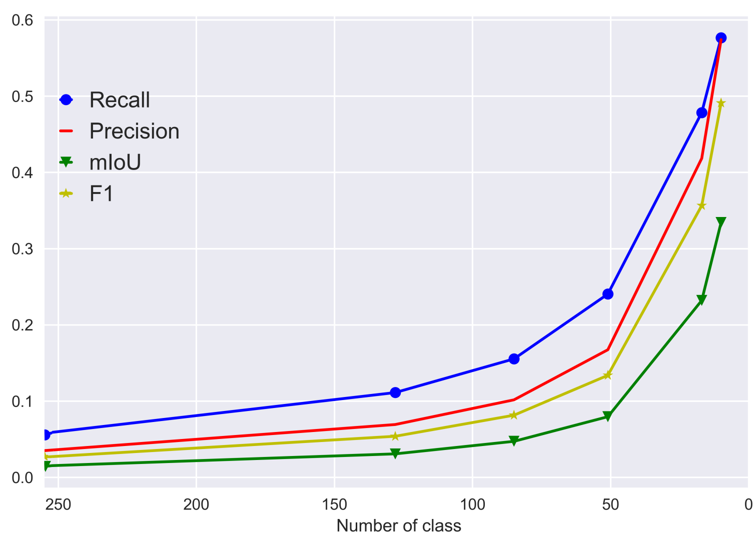


Figure 5: The Recall, Precision, mIoU, F1 score, are computed for different number of classes by fusion of adjacent classes.

convolutional neural network: the SegNet. The SegNet perform well and the result obtained are encouraging but despite the efficiency of this network, other architectures can be more appropriate and/or efficient. In fact, the artefact removal task can be, at numerous point, considered as noise reduction or in-painting.
From the point of view of Xie et al,[9] the de-noising is assumed as similar to in-painting and more precisely to blind in-painting. The blind in-painting does not have a specific given mask of the region to fill. The network is trained to identify the region to in-paint and the way to reconstruct it with correct semantic. One of the solution which can be adapted is the De-noising Auto-encoder (DA) as a base for a neural network called Staked Sparse DA.

ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of GPUs used for this research.

Figure 6: Illustration of results. Comparison between Input (top row), GT (middle row) and prediction (bottom row) of the SegNet network.

REFERENCES

[1] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." IEEE transactions on pattern analysis and machine intelligence 39.12 (2017): 2481-2495.

[2] Bergman, Ruth, et al. "Comprehensive solutions for automatic removal of dust and scratches from images." Journal of Electronic Imaging 17.1 (2008): 013010.

[3] Besserer, Bernard, and Cedric Thir. "Detection and tracking scheme for line scratch removal in an image sequence." European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2004.

[4] Zhou, Changyin, and Stephen Lin. "Removal of image artifacts due to sensor dust." 2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2007.

[5] Dirik, A. Emir, Husrev T. Sencar, and Nasir Memon. "Source camera identification based on sensor dust characteristics." 2007 IEEE Workshop on Signal Processing Applications for Public Security and Forensics. IEEE, 2007.

[6] Richard, Manuel M. Oliveira Brian Bowen, and McKenna Yu-Sung Chang. "Fast digital image inpainting." Appeared in the Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain. 2001.

[7] https://github.com/dstrub/DataSetFilmCorection

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

[9] Xie, Junyuan, Xu, Linli, et Chen, Enhong. Image denoising and inpainting with deep neural networks. In : Advances in neural information processing systems. 2012. p. 341-349.

[10] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).