Joint CNN and Variational Model for Fully-automatic Image Colorization

Thomas Mouzon, Fabien Pierre, Marie-Odile Berger

Joint CNN and Variational Model for Fully-automatic Image Colorization

Thomas Mouzon[1], Fabien Pierre[1], and Marie-Odile Berger[1]

[1]Laboratoire Lorrain de Recherche en Informatique et ses Applications
UMR CNRS 7503, Universite de Lorraine, INRIA projet Magrit, France

Abstract. This paper aims to couple the powerful prediction of the convolutional neural network (CNN) to the accuracy at pixel scale of the variational methods. In this work, the limitations of the CNN-based image colorization approaches are described. We then focus on a CNN which is able to compute a statistical distribution of the colors for each pixel of the image based on a learning stage on a large color image database. After describing its limitation, the variational method of [17] is briefly recalled. This method is able to select a color candidate among a given set while performing regularization of the result. By combining this approach with a CNN, we designed a fully automatic image colorization framework with an improved accuracy in comparison with CNN alone. Some numerical experiments demonstrate the increased accuracy reached by our method.

Keywords: Colorization, Convolutional Neural Networks, Total Variation, Optimization

1 Introduction

In video colorization, academic research has reached great improvement since the 1970s. In methodological terms, three types of approaches were proposed.

The first one is based on the diffusion of color points drawn by the user [13, 19]. These methods are based onto a diffusion of color with a local assumption: if the contours, so the gradients, are high for the luminance channel, they should be also high for the chrominance ones. The two drawbacks of this kind of methods are the tedious work needed by the user in case of complex images (for instance with textures), and the failure of diffusion method in case when there is not enough contrast to stop the diffusion of colors at the contours. The second category of approaches is based on a reference colored image [9] which is used as an example. In addition, the difficult problem of colorization based on example faces has been solved by calculating diffeomorphisms between images [15]. Segmentation and patch methods have been introduced to take into account example images. Some variational models for colorization that combine several results under the assumption of spatial and temporal regularity have been proposed [16, 17]. Nevertheless, whereas theses methods are fully automatic when the reference image is given, its choice may be critical.

Some models are based on the minimization of a cost function. They have been developed in order to regularize the results of colorization [14,17], both for example and manual methods.

The third colorization approach uses some large image databases [23]. Neural networks (Convolutional Neural Networks, Generative Adversarial Networks, Autoencoder, Recursive Neural Networks) have also been used successfully leading to a significant number of recent contributions. This literature can be divided into two categories of methods. The first evaluates the statistical distribution of colours for each pixel [3, 18, 23]. The network computes, for each pixel of the gray-scale image, the probability distribution of the possible colors. The second takes a grayscale image as input and provides a color image as output, mostly in the form of chrominance channels [1,5–8,10,12,21]. Some methods use a hybrid of both (*e.g.*, [24]).

Both techniques require image resizing, that is either done by deconvolution layers or performed *a posteriori* with standard interpolation techniques.

Target (input) Result of [23] Our model

Fig. 1. Example of halo effects produced by the method of [23]. Based on a variational model, our method is able to remove such artifacts.

In the case of [23], the network computes a probability distribution of the color on a down-sampled version of the original image. The choice of a color in each pixel at high resolution is made by linear interpolation without taking into account the grayscale image. Hence, the contours of chrominance and luminance may be not aligned, producing halo effects. Figure 1 shows some grey halo effects at the bottom of the cat that are visible on the red part, near the tail. In the other hand, in comparison to the others approaches of the state-of-the-art, the method of [23], produces images which are shiniest. We also show in this paper that we can make it a little bit shinier. Visually, the results of the competitive methods [8, 12] look drabber. In the following, the method of [23] is integrated in our system to predict colors.

In image colorization, convolutional neural networks can be used to compute in each pixel a set of possible colors and their associated probabilities [23]. However, since the final choice is made without taking into account the regularity of the image, this leads to halo effects. To improve this, we first propose to adapt the functional of [17] to the regularization of such results within the framework of colorization. The method of [17] being able to choose between

several color candidates in each pixel, it will be quite easy to use on the color distribution provided by the CNN described in [23]. In addition, the numerical results of [17] demonstrate the ability to remove halos, which is relevant to the limitations of [23]. This functional will have to face two main problems: on the one hand, the transition from a low to a high resolution, and on the other hand, the maintenance of a higher saturation than current methods.

In this paper, the CNN described in [23] is presented in the first section. In the second one, the functional of [17] is recalled. The next section describes the way to couple the methods of [23] and [17]. Finally, the last section shows some numerical comparisons with some state-of-the art methods.

2 A CNN to compute a statistical distribution of color

The method of [23] is based on a discretization of the CIE Lab color space into C=313 colors. This number of reference colors comes from the intersection gamut of the RGB color space and the discretization of the Lab space. The authors designed a CNN based on a VGG network [20] in order to compute a statistical distribution of the C colors in each pixel. The input of the network is the L lightness channel of the Lab transform of an image of size 256×256. The output is a distribution of probability over a set of 313 couples of a, b chrominance values for each pixel of a 64×64 size image. The quantification of the color space in 313 colors is computed from two assumptions. First, the colors are regularly spaced onto the CIE Lab color space. On this color space two colors are close with respect to the Euclidean norm when the human visual system feels them close. The second assumption that rules the set of colors is the respect of the RGB gamut. The colors have to be displayable onto a standard screen.

To train this CNN, the database ImageNet [4] is used without the gray-scale images. The images are resized at size 256×256 and then transformed into the CIE Lab colorspace. The images are then resized at size 64×64 to compute the a and b channels. The loss-function used is the cross-entropy between the luminance (a, b) of the training image and the distribution over the 313 original colors. Let us denote by Δ the probability simplex in C=313 dimensions.

Denoting by $(\hat{w}_i(x))_{i=1..C} \in \Delta^N$ the probability distribution of dimension C in the N pixels of the 64×64 image (over a domain Ω), and denoting by $(w_i(x))$ the ground truth distribution computed with a soft-encoding scheme (see [23] for details), the loss-function is given by:

$$L(\hat{w}, w) = -\sum_{x\in\Omega}\sum_{i=1}^{C} w_i(x)\log(\hat{w}_i(x)). \tag{1}$$

The forward propagation in the network provides a probability distribution over the C colors. In order to compute a colorization result, a choice among all these colors has to be performed. Basically, the authors of [23] proposed an annealed-mean in each pixel, independently. After that, a resizing of the (a, b)

channels at original size is done and recombined with brightness channel to obtain the color image.

Nevertheless, this recombination is done without taking into account any spatial consideration. In the next section we recall how the functional of [17] works to adapt it.

3 A Variational Model for Image Colorization with Channels Coupling

In [17], the authors have proposed a functional that selects a color among candidates extracted from a patch-based method. Assuming that C candidates are available in each pixel of a domain Ω and assuming that two chrominance channels are available for each candidate. Let us denote for each pixel at position x the i-th candidate by $c_i(x)$, $u(x) = (U(x), V(x))$ stands for chrominances to compute, and $w(x) = \{w_i(x)\}$ with $i = 1, \ldots, C$ for the candidate weights. Let us minimize the following functional with respect to (u, w):

$$F(u,w) := TV_{\mathfrak{C}}(u) + \frac{\lambda}{2}\int_{\Omega}\sum_{i=1}^{C} w_i \|u(x) - c_i(x)\|_2^2\, dx + \chi_{\mathcal{R}}(u(x)) + \chi_{\Delta}(w(x))\ . \tag{2}$$

The central part of this model is based on the term

$$\int_{\Omega}\sum_{i=1}^{C} w_i(x)\|u(x) - c_i(x)\|_2^2\, dx. \tag{3}$$

This term is a weighted average of some L2 norms with respect to the candidates c_i. The weights w_i can be seen as a probability distribution of the c_i. For instance, if $w_1 = 1$ and $w_i = 0$ for $2 \leq i \leq C$, the minimum of F with respect to u is equal to the minimization of

$$TV_{\mathfrak{C}}(u) + \frac{\lambda}{2}\int_{\Omega}\|u(x) - c_1(x)\|_2^2\, dx + \chi_{\mathcal{R}}(u(x)). \tag{4}$$

To simplify the notations, the dependence of each value to the position x of the current pixel will be removed in the following. For instance, the second term of (2) will be denoted by $\int_{\Omega}\sum_{i=1}^{C} w_i(x)\|u(x) - c_i(x)\|_2^2\, dx$.

This model is a classical one with a fidelity-data term $\int_{\Omega}\sum_{i=1}^{C} w_i\|u - c_i\|_2^2$ and a regularization term $TV_{\mathfrak{C}}(u)$. Since the first step of the method extracts many candidates, we propose averaging the fidelity-data term issued from each candidate. This average is weighted by w_i. Thus, the term

$$\int_{\Omega}\sum_{i=1}^{C} w_i\|u - c_i\|_2^2 \tag{5}$$

connects the candidate color c_i to the color u that will be retained. The minimum of this term with respect to u is reached when u is equal to the weighted average of candidates c_i.

Since the average is weighted by w_i, these weights are constrained to be onto the probability simplex. This constraint is formalized by $\chi_{\Delta}(w)$ whose value is 0 if $w \in \Delta$ and $+\infty$ otherwise, with Δ defined as:

$$\Delta := \left\{ (w_1, \cdots, w_C) \text{ s.t. } 0 \leq w_i \leq 1 \text{ and } \sum_{i=1}^{C} w_i = 1 \right\}. \tag{6}$$

Let $TV_{\mathfrak{C}}$ be a *coupled* total variation defined as

$$TV_{\mathfrak{C}}(u) = \int_{\Omega} \sqrt{\gamma \partial_x Y^2 + \gamma \partial_y Y^2 + \partial_x U^2 + \partial_y U^2 + \partial_x V^2 + \partial_y V^2}\ , \tag{7}$$

where Y, U and V are the luminance and chrominance channels. γ is a parameter which enforces the coupling of the channels. Some others total variation formulations have been proposed to couple the channels, see for instance [11] or [2].

In order to compute a suitable solution for problem (2), authors of [17] propose a primal-dual algorithm with alternating minimization of the terms depending of w. They also proposed numerical experiments showing the convergence of their algorithm. Let us note that this recent reference shows that the convergence of such numerical schemes can be demonstrated after smoothing of the total variation term. Among all the numerical schemes proposed in the references [17, 22], we choose the methodology having the best convergence rate as well as a convergence proof. This scheme is given in Algorithm 2 in [22]. This algorithm is a block coordinate forward backward algorithm. To increase the speed-up of the convergence, Algorithm 2 of [22] is initialized with the result of 500 iterations of the primal-dual algorithm of [17]. Whereas this algorithm has no guaranty of convergence, the authors of [22] have experimentally observed that it numerically converges faster.

Unfortunately, the functional (2) is highly non-convex and contains many critical points. More precisely, the functional is convex with respect to u with fixed w and reversely, it is convex with respect to w for fixed u. Nevertheless, the functional is not convex with respect to the joint variables (u, w). Thus, even if the numerical scheme would converge to a local minimum, the solution of the problem highly depends on the initialization.

In the next section, we will show how the powerful prediction of CNN can be used to tackle this last problem.

4 Joining Total Variation Model with CNN

In this section, a method to couple the prediction power of CNN with the precision of variational methods is described. To this aim, let us remark that the variable w of the functional (2) represents the ratio of each color candidate which

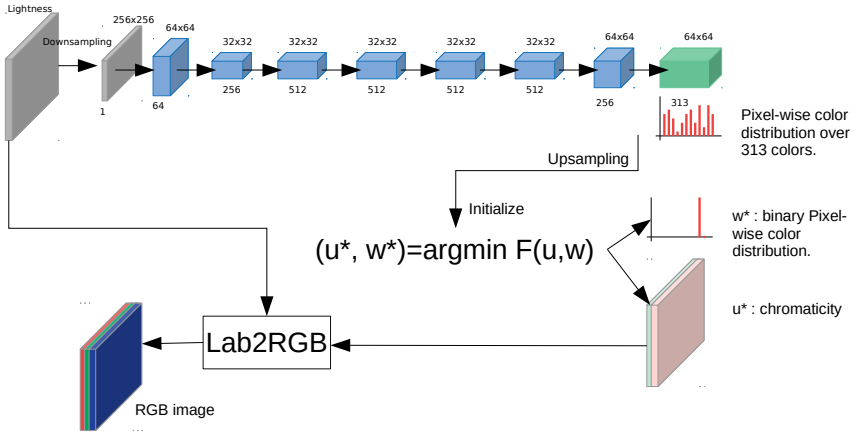


Fig. 2. Overview of our method. A CNN computes color distribution on each pixel. A variational method selects then a color for each pixel based on a regularity hypothesis.

is represented in the final result. This comes from the fact that, for a given vector $w \in \mathbb{R}^C$, the minimum of

$$\sum_{i=1}^{C} w_i \|u - c_i\| \tag{8}$$

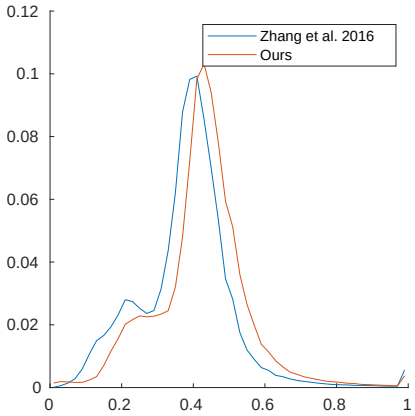
with respect to u is given by

$$\sum_{i=1}^{C} w_i c_i. \tag{9}$$

Thus, it can bee seen as a probability distribution of the colors in the desired color image, which is exactly the same purpose of the CNN in [23].

Figure 2 shows an overview of our method. First, the gray-scale image, considered as the luminance L is given as an input to the CNN. The output of the CNN is a probability distribution over 313 possible chromaticity at low resolution (64×64). In order to initialize the minimization algorithm, the output weights of the CNN can be used. The CNN provides a coarse scale output, that needs an up-sampling before producing a suitable output at original definition. Two ways can be considered. For the first one, the variational method can be used at coarse scale (low definition), and then an interpolation can be performed to recover a result at fine scale (high definition). For the second one, the probability distributions can be interpolated to get a high definition array. In the following, the second approach will be preferred. Indeed, the interpolation of a color image produces a decrease of the saturation, that makes images drabber. By interpolating the probability distributions instead of the color images, the variational method will be able to compute a color for each pixel based on a coupling of the channels at high resolution. The given probability distribution is then used as

Original image Zhang *et al.* 2016 Our result



Histograms of saturation

Fig. 3. Results of Zhang *et al.* [23] compared with our result. The histogram of the saturation shows that our result is shinier than the original method. Indeed, the average value of the saturation is higher for our model (0.4228) than the one of [23] (0.3802).

initialization value for the numerical scheme. As it was still proposed in [17], the variable u is initialized with $\sum_{i=1}^{C} w_i c_i$. After the iterations of the functional, the result, denoted by (u^*, w^*), provides some binary weights (see, *eg*, [17], Section 2.3.2) and a regularized result u^* that gives two chromaticity channels, a and b, at initial definition. Recombined with the luminance L and transformed into the RGB space, that produces a color image.

Let us remark that the authors of [23] proposed to first produce the color image and then to resize it with bi-cubic interpolation. Unfortunately, up-sampling or down-sampling images with bi-linear or bi-cubic interpolations reduce the saturation of the colors and make them drabber than the original. To avoid that, we propose here the opposite approach: we first up-sample the color distribution, and then we compute a color image at full definition by using it. Since the numerical scheme is used at full definition, the required memory of the algorithm for all the weights and the colors is a limitation to process high resolution images on a standard PC. To tackle this issue, we propose to select some of the

313 colors. This selection is done with respect to the probability distribution of the colors, by choosing the 10 highest modes.

This choice of 10 has been done experimentally. For most images, 8 or 9 candidates are enough and taking more of them does not improve the result, but it increases the computational time. On the other hand, taking less candidates decreases the quality of the result on a significant number of images. Finally, the number of 10 is a fair trade-off.

The training step of the CNN is done as in [23]. The variational step is not taken into account during the training process. Indeed, the relation between the initialization of the weights and the result is not analytically described and the gradient back-propagation algorithms is not suitable for this problem. Thus, the training is done by feeding the CNN with a gray-scale image as input and a color distribution as output. The variational step remains independent of the full framework during the training step. Its integration will be the purpose of future works.

In the next section, the numerical results are presented.

5 Numerical Results

Fig. 4. Comparison of our method with [23]. This example provides a proof of concept. Our method is able to remove the halo effects on the colorization result of [23].

In this section we show a qualitative comparison between [23] and our framework. A lot of results provided by [23] are accurate and reliable. We will show on these examples that our method does not reduce the quality of the images. We then propose some comparisons with erroneous results of [23], which shows that our method is reliable to fully automatically colorize images without artifacts and halo effects. A time comparison between the CNN inference computation and the variational step will be proposed to show that the regularization of the result is not a burden on the CNN approach. Finally, to show the limitation of CNN in image colorization, we will show some results where neither the approach of [23] nor our framework are able to produce some reliable results.

Fig. 5. Comparison of our method with [23].

Figure 3 shows the colorization results of the method of Zhang *et al.* [23]. Whereas it is hard to see that our method produces a shinier result that the result of [23] unless being a calibration expert, the histogram of the saturation is able to show the improvement. Indeed, since the histogram is right-shifted, it means that globally, the saturation is higher on our result. Quantitatively, the average of the saturation is equal to 0.4228 for our method, while it is equal to 0.3802 for the method of [23]. This improvement comes from the fact that our method selects one color among the ones given by the results of the CNN, whereas the method of [23] computes the annealed mean of them. The mean of the colors of the chrominances produces a decrease of the saturation and makes the colors drabber. By using a selection algorithm based on the image regularization, our method is able to avoid this drawback.

The result in Figure 4 is a proof of concept for the proposed framework. We can see a toy example which is automatically colorized by the method of [23]. The result given by the method of [23] produces some halo effect near the only contour of the image, which is unnatural. The regularization of the result is able to remove this halo effect and to recover an image looking less artificial. This toy example contains only two constant parts. The aim of the variational

Target (input) Result of [23] Our model

Fig. 6. Additional comparisons of our method with [23].

method is to couple the contours of the chrominance channels and the ones of the luminance. The result produced with our method contains no halo effect, showing the benefits of our framework.

In Figure 5, we show some results and we compare them to the method of [23]. For the lion, (first line), a misalignment of the colors with the gray scale image is visible (a part of the lion is colorized in blue and a part of the sky is brown beige). This is a typical case of halo effect where our framework is able to remove the artifacts. For the image of mountaineer, on the result of [23] some pink stains appear. With our method, the minimization of the total variation ensures the regularity of the image, thus it removes these strains.

Figure 6 shows additional results. The first line is an old port-card. Its colorization is reliable with the CNN and, in addition, the variational approach makes it a little bit shinier. This example shows the ability of our approach to colorize historical images. In the second example, most of the image is well colorized by the original method of [23]. Nevertheless, the lighthouse as well as the right-side building contain some orange halos that are not reliable. With the variational method, the colors are convincing. Additional results are available on `http://www.fabienpierre.fr/ssvm2019`

The computational time of the CNN forward pass is about 1.5 sec in GPU, whereas the minimization of the variational model (2) is about 15 sec in Matlab in CPU. In [16], the authors provide a computation time almost equal to 1 sec with unoptimized GPU implementation. Since the minimization scheme of [22] is about the same, the computational time would be almost equal. Thus, the

computational time of our approach is not a burden in comparison with the method of [23].

In Figure 7, a failure case is shown. In this case, since the minimization of the variational model strongly depends on its initialization, our method is not able to recover realistic colors. Actually, fully automatic colorization remains an open problem.

Target (input) Result of [23] Our model

Fig. 7. Fail case. The prediction of the CNN is not able to recover a reliable color.

Conclusion

In this paper, we propose a novel approach to couple the power of the CNN with the precision of the variational models. This coupling is done with a transfer of information based on probability distributions. The computation of the two parts of the framework is based on standard techniques issued from the literature. The numerical results show the improvement of colorization results performed with the two methods considered together. Some results where neither the approach of [23] nor our framework are able to produce some reliable results are presented. Thus, image colorization remains an open issue despite the huge number of CNN-based approaches proposed in state-of-the-art.

References

1. Cao, Y., Zhou, Z., Zhang, W., Yu, Y.: Unsupervised diverse colorization via generative adversarial networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 151–166. Springer (2017)
2. Caselles, V., Facciolo, G., Meinhardt, E.: Anisotropic cheeger sets and applications. SIAM Journal on Imaging Sciences 2(4), 1211–1254 (2009)
3. Chen, Y., Luo, Y., Ding, Y., Yu, B.: Automatic colorization of images from chinese black and white films based on cnn. In: 2018 IEEE International Conference on Audio, Language and Image Processing. pp. 97–102 (2018)

4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009)
5. Deshpande, A., Lu, J., Yeh, M.C., Chong, M.J., Forsyth, D.A.: Learning diverse image colorization. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2877–2885 (2017)
6. Guadarrama, S., Dahl, R., Bieber, D., Shlens, J., Norouzi, M., Murphy, K.: Pixcolor: Pixel recursive colorization. In: British Machine Vision Conference (2017)
7. He, M., Chen, D., Liao, J., Sander, P.V., Yuan, L.: Deep exemplar-based colorization. ACM Transactions on Graphics 37(4), 47:1–47:16 (Jul 2018)
8. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. ACM Transactions on Graphics 35(4) (2016)
9. Irony, R., Cohen-Or, D., Lischinski, D.: Colorization by example. In: Eurographics Symp. on Rendering. vol. 2. Citeseer (2005)
10. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
11. Kang, S.H., March, R.: Variational models for image colorization via chromaticity and brightness decomposition. IEEE Transactions on Image Processing 16(9), 2251–2261 (2007)
12. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: European Conference on Computer Vision. pp. 1–16. Springer (2016)
13. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. In: ACM Transactions on Graphics. vol. 23–3, pp. 689–694 (2004)
14. Lézoray, O., Ta, V.T., Elmoataz, A.: Nonlocal graph regularization for image colorization. In: IEEE International Conference on Pattern Recognition. pp. 1–4 (2008)
15. Persch, J., Pierre, F., Steidl, G.: Exemplar-based face colorization using image morphing. Journal of Imaging 3(4), 48 (2017)
16. Pierre, F., Aujol, J.F., Bugeau, A., Ta, V.T.: Interactive video colorization within a variational framework. SIAM Journal on Imaging Sciences 10(4), 2293–2325 (2017)
17. Pierre, F., Aujol, J.F., Bugeau, A., Papadakis, N., Ta, V.T.: Luminance-chrominance model for image colorization. SIAM Journal on Imaging Sciences 8(1), 536–563 (2015)
18. Royer, A., Kolesnikov, A., Lampert, C.H.: Probabilistic image colorization. In: British Machine Vision Conference (2017)
19. Sapiro, G.: Inpainting the colors. In: IEEE International Conference on Image Processing. vol. 2, pp. II–698 (2005)
20. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015)
21. Su, Z., Liang, X., Guo, J., Gao, C., Luo, X.: An edge-refined vectorized deep colorization model for grayscale-to-color images. Neurocomputing (2018)
22. Tan, P., Pierre, F., Nikolova, M.: Inertial alternating generalized forward-backward splitting for image colorization. Journal of Mathematical Imaging and Vision (2019)
23. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European Conference on Computer Vision. pp. 1–16. Springer (2016)
24. Zhang, R., Zhu, J.Y., Isola, P., Geng, X., Lin, A.S., Yu, T., Efros, A.A.: Real-time user-guided image colorization with learned deep priors. ACM Transactions on Graphics 9(4) (2017)