Estimation of Scribble Placement for Painting Colorization

Cristian Rusu and Sotirios A. Tsaftaris

IMT Institute for Advanced Studies, Lucca, Italy

{cristian.rusu, s.tsaftaris}@imtlucca.it

Abstract—Image colorization has been a topic of interest since the mid 70's and several algorithms have been proposed that given a grayscale image and color scribbles (hints) produce a colorized image. Recently, this approach has been introduced in the field of art conservation and cultural heritage, where B&W photographs of paintings at previous stages have been colorized. However, the questions of what is the minimum number of scribbles necessary and where they should be placed in an image remain unexplored. Here we address this limitation using an iterative algorithm that provides insights as to the relationship between locally vs. globally important scribbles. Given a color image we randomly select scribbles and we attempt to color the grayscale version of the original. We define a scribble contribution measure based on the reconstruction error. We demonstrate our approach using a widely used colorization algorithm and images from a Picasso painting and the peppers test image. We show that areas isolated by thick brushstrokes or areas with high textural variation are locally important but contribute very little to the overall representation accuracy. We also find that for the case of Picasso on average 10% of scribble coverage is enough and that flat areas can be presented by few scribbles. The proposed method can be used verbatim to test any colorization algorithm.

I. Introduction

Colorization is a computer-assisted process of adding color to a monochrome image or movie. In recent years, the literature of colorization methods/algorithms has grown substantially. A widely known approach is introduced in [1], based on the assumption that neighboring pixels in an image that have similar intensities will also have, most likely, the same color. The problem is turned into a quadratic convex optimization problem and solved using efficient techniques. Thus, this class of algorithms requires the grayscale image and color scribbles (initial color hints) to be spread. In an attempt to speed up the process, [2] introduces a new local distance definition and reduces the color confusion between two distinct regions of the image. Another approach to the colorization problem is described in [3] based on the concepts of luminance-weighted chrominance blending and intrinsic geodesic distance computations that can be efficiently and accurately done. Taking ideas from the machine learning community, unsupervised formulations of the colorization problem have also been proposed [4], [5]. In this setting, given a colored image, called source, and a grayscale image, called destination, the goal is to transfer the colors from source to destination in a natural way.

However, until recently most colorization efforts were focused towards colorizing digital photographs and films. In the past two centuries art historical and conservation research have benefited from information revealed particularly from the study of historical photographs of works of art in their final or intermediate states. They provide a valuable tool for gaining knowledge on the artwork's evolution and the artist's creative path. Obviously, until the widespread introduction of color photography these photos were usually black and white (B&W) and the lack of color information can limit a full understanding of the portrayed pieces, both from an art-historical and conservation perspective.

Initial attempts at adding color to B&W images in a museum setting involved a hand-colored replica. Nowadays conservators, after digitizing the B&W photograph, typically rely on commercial software to digitally paint over the image. With this approach, the user utilizes several digital photo-editing tools, to color the B&W photograph. The process requires significant time and manual effort, and unless special techniques are employed (transparent layers), adjusting the color tone to the underlying B&W intensity is not trivial and the brushstrokes of the original work are hardly preserved. This has led to the introduction of digital colorization in a museum setting for the purpose of colorizing archival B&W photographs of paintings. Adapting a method from the literature, it was shown in [6], [7], [8] that colorizing B&W photographs of paintings is possible and beneficial to the cultural heritage preservation community.

One of the difficulties in using scribble based colorization algorithms is the need to have available scribbles, which are commonly given by the user. Baring color transfer methods, which may not always be applicable, it is critical to identify strategies that reduce the number of scribbles needed by the colorization method or given a colorization method identify better placement of scribbles. In [9], in an attempt to reduce the number of necessary scribbles, the colorization procedure is split into two stages: the color labeling, that groups pixels that might have the same color into coherence regions, and the color mapping, that improves and fine-tunes the colors in each defined region. More recently, techniques from sparse representations were also applied to the colorization problem in an attempt to further reduced the amount of human input[10]. However, the problem of given an algorithm what is the optimum scribble placement and their number remains unexplored.

This can be largely explained since the problem of finding (estimating) the minimum number of scribbles and their optimal positions in preparation for the application of a colorization algorithm is a difficult non-convex, discrete optimization problem (ℓ_0 pseudo-norm problem [11]). Even worse, since these algorithms are applied on image data of large dimensions the solution space is vast. These factors make the quest for a good solution rather difficult.

Here we address this limitation using an iterative algorithm

that provides insights as to the relationship between locally important scribbles versus scribbles that contribute to the overall result. Given a color image we randomly select scribbles and we attempt to color the grayscale version of the original. We define a scribble contribution measure based on the reconstruction error, considering local and global contributions, and we iteratively repeat this random process. At each iteration, we record a scribble importance map and we use this map to drive the probability of how each pixel is selected. Thus, the method converges after a few iterations. We demonstrate our approach using a widely used colorization algorithm on images from a Picasso painting and a popular test image. We show that areas isolated by thick brushstrokes or areas with large texture variations are locally important but do not contribute to the overall representation accuracy. We also find that for the Picasso painting on average 10% of scribble coverage is enough and that flat areas can be presented by few scribbles. The two experimental settings, a painting and a natural image, show that the proposed method can be successfully applied to various image types.

The rest of the paper is organized as follows, in Section II we briefly outline the colorization algorithm employed here and we describe in detail the proposed numerical and iterative method for estimating scribble placement. In Section III we present results based on a known painting of Picasso and discuss the outcomes of the proposed approach. Finally, in Section IV we offer conclusions.

II. PROPOSED METHOD

In this section we first briefly describe the colorization approach that we follow and subsequently we describe the process for iteratively estimating scribble importance.

A. Colorization approach

Here we adopt the methodology in [1], although the approach presented in the following section works with any colorization algorithm. This optimization-based colorization algorithm, aims at colorizing an image using a few predefined scribbles [1]. Assuming a YUV color space our goal was to estimate the U and V components using only the grayscale, luminance or intensity component Y, which in our case is the Druet image, and scribbles, which are described in a following section. Since the procedure is the same for both U and V components, only the procedure for U is presented.

Briefly described, the method in [1] relies on the fact that the color information at specific locations (termed scribbles) is defined *a priori* by the user (and possibly via color transfer approaches [6]). The basic premise of the algorithm is that pixels with similar luminance values will have similar chrominance values. To accomplish this goal a quadratic optimization problem is cast with respect to the pixels with unknown color and the solution provided by solving a simple, but large, sparse linear system of equations. More specifically, for colorizing the U chrominance component the following cost function is minimized with respect to the un-colorized pixels $U(\mathbf{s})$

$$J(U) = \sum_{\mathbf{r}} \left(U(\mathbf{r}) - \sum_{\mathbf{s} \in N(\mathbf{r})} w_{\mathbf{rs}} U(\mathbf{s}) \right), \qquad (1)$$

where vectors $\mathbf{r}, \mathbf{s}$ denote pixel locations, $U(\mathbf{r})$ the known values (scribbles), $U(\mathbf{s})$ the unknown values, $N(\mathbf{r})$ the neighborhood of $\mathbf{r}$ and $w_{\mathbf{rs}}$ a weight function determining the degree of similarity of $U(\mathbf{r})$ and $U(\mathbf{s})$. This weight function is defined based on the luminance values so as to enforce similarity in color based on similarity in intensity. Here as suggested in [1], we use

$$w_{\mathbf{rs}} \propto \exp\left(-\frac{(Y(\mathbf{r}) - Y(\mathbf{s}))^2}{2\sigma_{\mathbf{r}}^2} \right), \qquad (2)$$

where $\sigma_{\mathbf{r}}$ is the variance of intensities in the neighborhood $N(\mathbf{r})$, where we used a 3×3 square neighborhood. Minimization of (1) results in the solution of the linear set of equations

$$\mathbf{Au} = \mathbf{b}, \qquad (3)$$

where $\mathbf{u}$ is the vector of the unknown chrominance components, $\mathbf{b}$ the vector of the known values of the chrominance at the predefined locations of the scribbles and zeros otherwise; and $\mathbf{A}$ is a large square sparse matrix in which every row represents the relationship of each pixel with its neighbors as defined by the weight function $w_{\mathbf{rs}}$. For rows corresponding to pixels of unknown color, the row sums to one and has entries on the N neighborhood of the pixel; otherwise for pixels with known color (scribbles) the row has a single one in the diagonal such that (3) reduces to identity for that pixel. The solution of (3) provides the U component for the image and the process is repeated for the V component.

B. Scribble importance estimation

It is evident from the description of the colorization method in the previous section, that the important choice of $\mathbf{b}$ falls to the user. Our goal is to help the user in making this choice such that manual effort is reduced without affecting the results of the colorization algorithm. In the following we give a detailed description of the proposed iterative scribble estimation algorithm.

In order to test the performance of the proposed method we assume that we also have available the original colored image. This allows us to automatically set and test the importance of scribbles deployed in various regions. The grayscale image is of course also supplied as input.

The main idea of the algorithm is to randomly make scribble selections of fixed coverage (ie., how many pixels in the image we have knowledge of their color) and test the result of the colorization. (We are interested to see how well a certain group of randomly selected scribbles reconstructs the original color image.) We apply this check in an iterative manner where each time we pick a random subset of image pixels (all pixels are candidates to become scribbles). The probabilities of picking each pixel are altered following a Reactive Search Optimization (RSO) [12] approach, which allows us to sample a large solution space and converge faster to a local optimum. We compute an error index for each selected scribble in the current step. This error index involves the smoothed absolute error in percentages of both the global and local reconstruction accuracy given the current selected scribbles.

The details of the algorithm are presented next.

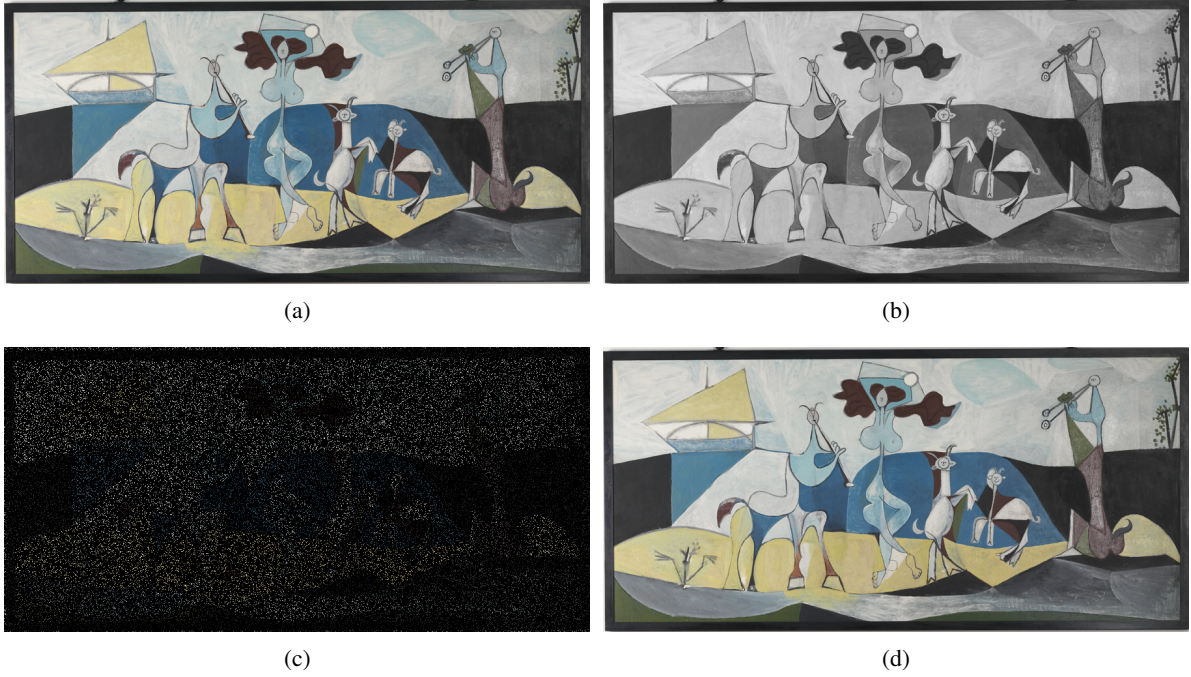**eScribbles - Iterative Estimation of Scribble importance using an RSO strategy.** Given the grayscale image

Fig. 1: (a) La Joie de Vivre (LJDV) by Pablo Picasso, 1946 (Musée Picasso, Antibes France, 1946.1.4; oleoresinous enamel paint and charcoal on fibrocement, 120×250 cm); (b) LJDV in grayscale; (c) an example of randomly selected scribbles covering 10% of the image; and (d) the result of colorizing (b) using the scribbles in (c) and the colorization method described in text.

$I_{\text{gray}} \in \mathbb{R}^{m \times n}$, its original colored version $I \in \mathbb{R}^{m \times n \times 3}$, the maximum number of iterations K, a stopping threshold T and a percentage ρ of scribble coverage the algorithm returns a scaled probability map that highlights the importance of each image pixel.

- Iterations: for $k = 1, \ldots, K$

1) Selection step:
   a) If $k < \lceil K/3 \rceil$ pick uniformly at random ρ% of the available pixels.
   b) Otherwise pick randomly ρ% of the available pixels by using the currently computed importance map to get a probability measure. The probability measure taken is given by $P = f_B(\tilde{\mathcal{E}})$, where f_B is a function that first scales $\tilde{\mathcal{E}}$ to be in the range of [0,1] and then uses a per-pixel Bernoulli process to decide which pixel to choose.
   Denote as $\mathbf{S}$ the set of pixel locations (i, j) of each selected pixel.

2) Turn the selected pixels into scribbles by reading their chrominance information from the colored image I.

3) Using the new scribbles, solve the colorization problem as defined in (2) (3) by updating $\mathbf{A}, \mathbf{b}$ and colorize I_{gray} to obtain the new reconstructed I' (the recolored version of I).

4) Compute the error E_H, as $E = |I - I'|$ and $E_H = E \circ H$, where I is the original image, I' is the reconstructed one, H is an averaging filter of radius R, and $\circ$ denotes 2D convolution.

5) For a scribble at location (i, j), we compute its normalized error contribution as

$$\mathcal{E}(i, j) = \frac{\sum_{(i,j) \in \mathbf{s}} E_H(i, j)}{E_H(i, j)}, \qquad (4)$$

its error normalized by the total error of the current scribble group.

6) Add the errors to a running average cumulative total error $\tilde{\mathcal{E}}$. This map describes the importance of every pixel as a scribble candidate.

7) Denote $\tilde{\mathcal{E}}_k$ and $\tilde{\mathcal{E}}_{k-1}$ the cumulative total errors at step k and $k-1$ respectively. If $\|\tilde{\mathcal{E}}_k - \tilde{\mathcal{E}}_{k-1}\|_F < T$ we consider that the algorithm has converged and we stop the iterations.

Step (1) of the iteration begins by picking points in a uniformly random (Monte Carlo) fashion to approximate the scribble importance of every pixel. After one third of the iterations have executed we reach the confidence level to use the current importance map as a probability map when taking into account the chances of picking the new scribbles. The first part allows for the algorithm to have a large spread of possible scribbles in the image while in the second part, it focuses on the pixels that are considered the most important and refines the result. This mechanism allows the algorithm to converge to a stable importance map in all tested experimental runs.

Steps (2-3) apply the actual colorization algorithm, with scribbles chosen from the colored image. Steps (4-5) of the iterations concern the calculation of an error that is indicative of the quality of each scribble in the current group. In step (6) the results are stored in a running average, while step (7) checks for the convergence of the algorithm.

III. RESULTS AND DISCUSSION

For our experiments we use a digital photograph from *La Joie de Vivre* in the Picasso collection of the Antibes Museum, shown in Figure 1a. The Joy of Life (La Joie de Vivre) depicts Francoise Gilot (Picasso's companion at the time) as the central nymph dancing in front of a paradise-like landscape of water and earth, accompanied by a faun and a centaur playing the double flute and two small goats. Picasso worked on this painting for almost two months and this is confirmed by numerous photographs taken by Michel Sima, a local artist and friend of Picasso. The colorization of these photographs was the subject of the study described in [7], [8].

We used the image in Figure 1a as our "ground truth" colored image, I, and image in Figure 1b as the grayscale image I_{gray} an inputs to the process described in Section II. To obtain results faster we use a downsampled version (900×440 pixels) of the original image.

Figure 1c shows an example image of uniformly sampled scribbles shown on black background that were used to color the input grayscale image. Figure 1d shows a colorized output obtained with the algorithm in Section II using the scribbles of the previously described image. Observe that the colorization output is very similar to the original image and to the naked eye the result is very satisfactory. This demonstrates the capability of this optimization based colorization algorithm in providing good colorized renditions of paintings (and natural images as well).

In all our experimental runs, the maximum number of iterations is set to $K = 6000$, the radius of the averaging operator is set to $R = 5$ and the stopping threshold $T = 10^{-5}$.

To identify a good choice for ρ we performed an approach similar to the one described in Section II. However, rather than keeping the amount of pixels ρ that are sampled at each iteration fixed, we dynamically increase them in the range of 0.01 to 0.2 and we record the average of E.

In Figure 2 we show the effect of ρ (ie., the percentage of pixels in the image of known color) on the corresponding error. Evidently the more scribbles the lower the error (which is expected). We can see though that for $\rho > 0.1$, the error falls below 1.7%. (Effectively, this translates that on average we are only $\approx 2\%$ off from the original color value, using only 10% of hints, and in some sense this process of finding a good ρ also identifies the amount of scribbles necessary for a given image). For the remainder a value of $\rho = 0.1$ was chosen for our experiments.

Overall the method converges very quickly past the first ($K/3$) uniformly drawn pixels; in fact, even given our strict convergence criteria convergence was observed after reaching approximately 1000 iterations. In Figure 3 we show the resulting scribble error index for all pixels in the Picasso image. This map identifies the importance of each pixel in the image. Low values show regions that if a scribble is placed there has low global contribution to the resulting error although it may locally important. High values on the other hand show that placing a scribble there will have a large contribution in the reconstruction. In essence, regions of high value correspond to regions of flat areas with uniform color. As expected, placing scribbles on large flat areas will contribute more towards
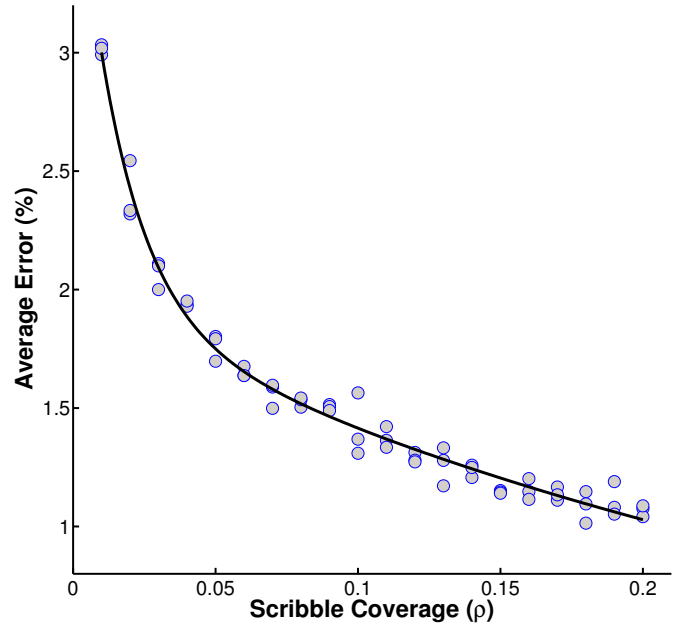

Fig. 2: Monte Carlo analysis to show the effect of ρ on the average absolute error between the colorized image and the original colored one. Each data point reflects an experiment with a random set of scribbles (of a given total coverage) and the corresponding error (shown as percent of the color value of the color-space). Within the same graph we show an exponential line fit for reference.

reducing the error. While regions of low value indicate isolated regions, ie., regions that are delineated with thick brushstrokes. For example, considering the sail of the boat shown in the upper left corner, scribbles should be placed on the inside of the triangle shape.

This map provides a visual guideline as to how scribbles should be placed. Few scribbles need to be placed in large flat regions to obtain an overall colorization; while if details need to be recovered additional scribbles are necessary within small detail carrying regions (such as the flute of the centaur in between the boat and the nymph). Thus, the user can use this information as a guideline to decide where to prioritize scribble placement for a given desired coverage. For example, Figure 4 shows the scribbles drawn by an expert for one of the archival photos of LJDV described in [7], [8]. Observe, that the expert drew several redundant lines within the sail of the boat for example; while, our analysis indicates that (albeit very important to indicate) much fewer scribbles are necessary and their exact placement within the sail is not so important. This observation extends to all flat regions of the image, thus greatly reducing human effort.

In order to demonstrate that our method extends well also to other types of digital images, in the following analysis, we apply our algorithm on a popular test image, the peppers image, shown in Figure 5a.

The map obtained with our method shown in Figure 5b clearly highlights the regions of interest, from a scribble
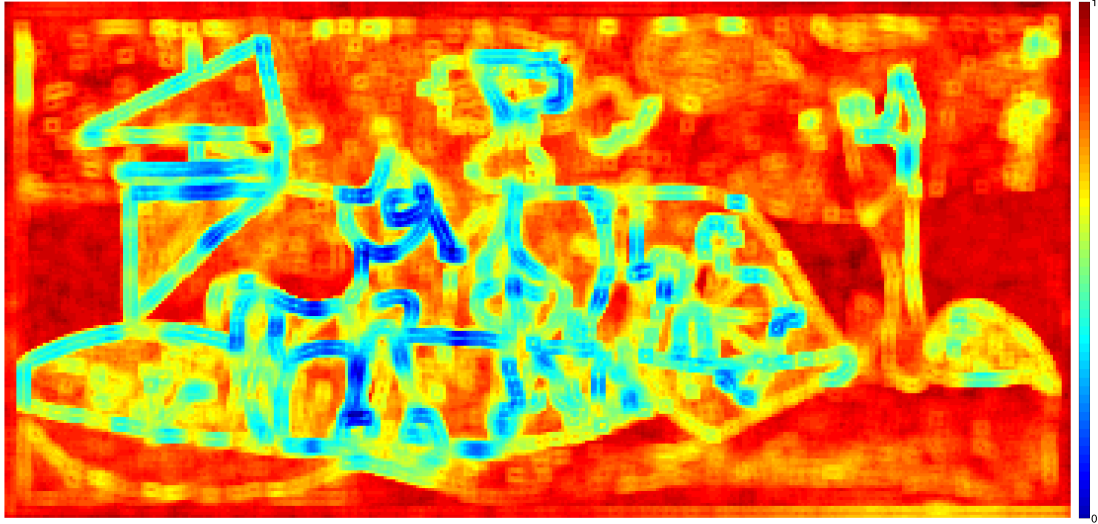
Fig. 3: The map obtained using the proposed iterative method for LJDV. Values are scaled and pseudo-colored using the shown colorbar and give the measure of scribble importance of each pixel.

Fig. 4: An example of scribbles as drawn by an expert in painting conservation.
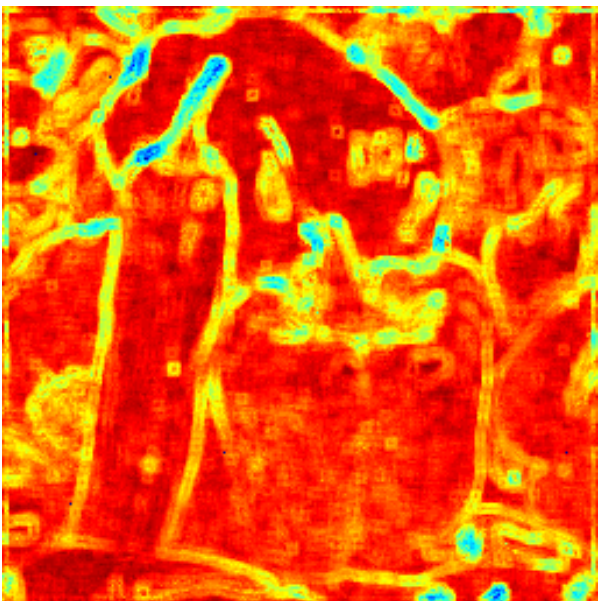
perspective, in the picture: the stems of the peppers, the green pepper almost entirely obscured by the red pepper in the upper-right section, the group of three small peppers (one of different color) in the upper-right corner. Since the rest of the image is composed of relatively large flat areas, where the precise scribble placement is not important, our method suggests that no colorization issues will be encountered there. Similarly to the previous example, most regions of the image are easily colored with the exception of a few, which require special attention. Indeed, the human generated scribbles for the pepper image in [3] follow the general map described by our method.

Although here we used a specific colorization algorithm, the approach presented here can be employed for any colorization method and will result in the scribble importance

map for that method. Thus, it can represent a tool for the colorization community in order to help would-be users in deciding where to place scribbles. Most of the methods discussed in the introduction (with the exception of possibly [13]) are computationally intensive and cannot offer real time operation (where the user can interactively see the results of colorization) particularly with images of large dimension (such as the ones of paintings). Thus, most non expert users in image processing and colorization cannot judge where to place scribbles. Our experience working with experts in the cultural heritage preservation domain, pointed to the need to explain the inner workings/limitations of the algorithm in order to help them in drawing scribbles. However, we believe that using some simple example images drawn from

(a)

(b)

Fig. 5: (a) Peppers test image; (b) Scribble error index provided by our algorithm.

the domain of images to be colored (for example images of other paintings created from the same artist around the same era, or the final state of the painting) and the corresponding scribble maps would enhance the interaction with the users and increase the deploy-ability of colorization methods (and other image restoration approaches) to the broad art conservation and history community.

IV. CONCLUSIONS

Although image colorization has been a topic of study in the image restoration community for a few decades the literature is lacking in studies that address where scribbles (color hints) should be efficiently placed. This paper addresses this limitation by exploring in an iterative fashion scribble placement. We test our approach in two applications: a Picasso painting and a popular test image, namely peppers. We found that scribbles should be placed in flat regions of large coverage for maximum return, as common intuition would suggest. Scribble placement is indeed laborious and our finding can aid in the design of strategies that guide the user of where to place scribbles and as such minimize even further user interaction. Unfortunately, our method does require an original color image to drive conclusions. However, in the context of painting colorization of archival photos usually the present state color painting is available. In other applications, such as the colorization of natural or photographic images, this limitation can be circumvented by identifying similar images in content from large image repositories (a strategy commonly adopted in color transfer methods).

REFERENCES

[1] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Transactions on Graphics*, vol. 3, pp. 689–694, 2004.

[2] Y. Li, M. Lizhuang, and W. Di, "Fast colorization using edge and gradient constrains," in *Proc. of 15th Int. Conf. WSCG*, 2007, pp. 309–315.

[3] L. Yatziv and G. Sapiro, "Fast image and video colorization using chrominance blending," *IEEE Trans. Image Processing*, vol. 15, no. 5, pp. 1120–1129, 2006.

[4] T. Welsh, M. Ashikhmin, and K. Mueller, "Transferring color to greyscale images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 277–280, 2002.

[5] D. Sykora, J. Burianek, and J. Zara, "Unsupervised colorization of black-and-white cartoons," in *Proc. 3rd Int. Symp. Non-Photorealistic Animation and Rendering*, 2004, pp. 121–127.

[6] S. A. Tsaftaris, K. H. Lister, I. Fiedler, F. Casadio, and A. K. Katsaggelos, "Colorizing a Masterpiece [Applications Corner]," *Signal Processing Magazine*, vol. 28, no. 3, pp. 113–119, 2011.

[7] S. A. Tsaftaris, F. Casadio, G. Gautier, J.-L. Andral, and A. K. Katsaggelos, "La joie de vivre: The evolution of a masterpiece," in *Picasso Express*, M. Raeburn and N. Radeuil, Eds. Antibes, France: Picasso Museum, 2011.

[8] ——, "A novel visualization tool for art history and conservation: automated colorization of black and white archival photographs of works of art," *Studies in Conservation*, in press.

[9] Q. Luan, F. Wen, D. Cohen-Or, L. Liang, Y. Xu, and H. Shum, "Unsupervised colorization of black-and-white cartoons," in *Proc. Eurographics Symp. Rendering Techniques, Grenoble, France*, 2007.

[10] K. Uruma, K. Konishi, T. Takahashi, and T. Furukawa, "Image colorization based on the mixed l0/l1 norm minimization," in *Proc. ICIP*, 2012, pp. 2113–2116.

[11] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 14, pp. 342–351, 2005.

[12] R. Battiti and T. Gianpietro, "The reactive tabu search," *ORSA Journal on Computing*, vol. 6, no. 2, pp. 126–140, 1994.

[13] P. Lagodzinski and B. Smolka, "Interactive colorization based on hybrid distance transform," in *In Proc. of CSAE*, vol. 3, 2011, pp. 538–543.