A Review of Colorization Methodologies

Authors
Viswajith Menon (vm623)
Zeeshan Ahsan (za224)

Abstract

Color plays a significant role in how humans process the world. Using rich, accurate and vivid colors helps in the process of human cognition. Colorization involves the task of automatically adding of colors to grayscale images and videos. This has numerous practical applications, including restoration of old photographs and movies, improvement of medical imaging, and creation of realistic animations and art. In recent years, significant progress has been made in developing various colorization methodologies that aim to automatically produce color images or videos from their grayscale counterparts. This literature review's goal is to examine cutting-edge methods for colorization in computer vision, including methods based on deep learning, methods based on optimization, and hybrid methods that combine these two approaches. The essential ideas and difficulties in colorization, such as color space representation, color transfer, and colorization accuracy assessment, will be covered in the survey. We intend to evaluate the benefits and drawbacks of various colorization methods and talk about how they might be used in the arts, sciences, and other sectors. Overall, this literature survey aims to provide a comprehensive understanding of the current state-of-the-art in colorization in computer vision and identify the research trends and opportunities for future exploration.

1 Background

Initial attempts at colorization were primarily manual. However, these methods were time-consuming and required a significant amount of expertise. The advancements of digital image processing, lead the way to the development of various colorization techniques that aimed at converting a greyscale photo/video into colored ones. Over the years, colorization has moved from being manual to semi-automatic to completely automatic which requires no human intervention. The different stages of colorization overtime –

1. Semi-Automatic
2. Automatic
3. Deep Colorization.

2 Semi-Automatic Colorization

The technique used in computer vision to add color to grayscale images with the help of some human input is known as semi-automatic colorization.

Source - target : Color transfer methods [18] [17] [1] are mainly employed to implement this strategy. Color transfer methods use the help of a source image which is colored. This source image is then used as a reference to add color to a greyscale image. This method yields good results when the source and target images share similar characteristics like contrast and brightness. However, this method is labor intensive as the source and target images must be manually matched.

Luminance-key based method : A luminance keying-based method for transferring color to a grayscale image is described in [6]. Color and grayscale values are matched with a pre-defined look-up table. The drawback of this system is that, when assigning different colors for a same gray level, a few luminance keys should be simultaneously manipulated by the user for different regions, making it a tiresome process

Optimization : These methods are used to find the best set of colors to apply to a grayscale image based on a specific objective or criteria. The paper [10] describes a method which allows users to create a few scribbles, with the observation that neighboring pixels in space-time share similar intensities should have similar colors. They then formulate colorization as an optimization problem for a quadratic cost function. The paper [70], presented a similar approach to [10]. However, this method achieved comparable colorization quality with [10] with much less computation time by quadtree decomposition based non-uniform sampling. This method greatly reduced the problem of color diffusion among different regions via designing a weighting function to represent intensity similarity in the cost function. The colorization technique presented in the paper [16], utilizes a local correlation-based optimization algorithm. However, its practical usage is restricted due to the requirement of color correlativity between pixels in various regions.

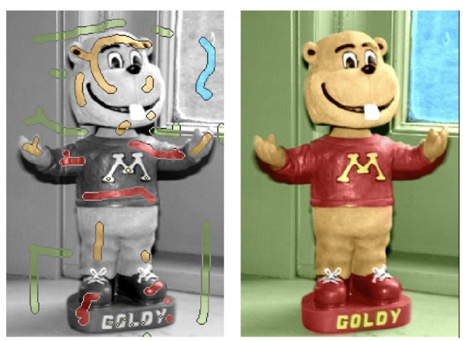


Figure 1: Colorization using optimization

The above image is an Illustration of the colorization method using optimization [10]. From left to right: an input grayscale image marked with color scribbles by the userand the the colorization result by [10].

Colorization method by transferring color from an exampler image is described in [8]. This paper uses a higher-level context of each pixel instead of making independent pixel-level decisions. This helps to achieve a better spatial consistency in the colorization result. With a supervised classification scheme, they estimate the best example segment for each pixel to learn color from. Then, by combining a neighborhood matching metric and a spatial filter for high spatial coherence, each pixel is assigned a color from the corresponding region in the example image. This approach requires considerably less scribbles than previously mentioned colorization methods

Chrominance blending – The paper [19] developed an image colorization method using chrominance blending. This method is based on the concept of color blending derived from a weighted distance function. This distance function is computed from the luminance channel. This method is fast and allows the user to interactively colorize a grayscale image by providing a reduced set of chrominance scribbles. This method is equally effective when implementing a brightness change or image recolorization.

PCA based – The paper [1], implemented grayscale colorization by applying a PCA (Principal Component Analysis) based transformation. The methods described, makes use of the fact that image color can be viewed as a highly correlated vector space. These colorizing methods then generate the

Figure 2: Colorization using Chrominance blending

color vector corresponding to the grayscale as a function. This method is significantly faster than previous approaches while producing visually acceptable colorization results. However, this method is restricted by complicated image segmentations that need to be executed by the user before applying PCA.

Color labeling and color mapping – [15] proposes an interactive method where pixels that should have similar colors are grouped into coherent regions. Color mapping is then applied to the grouped regions to generate a vivid colorization effect through assigning colors to a few pixels in the region. Every method described before this struggles in handling the task of colorizing a texture for any given color space. The method described in [15] handles texture by grouping both neighboring pixels sharing similar intensity and remote pixels with similar texture. This method proves extremely useful when colorizing images where different textures of a color are used.

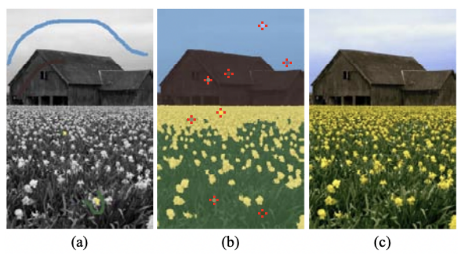


Figure 3:

First, several strokes with pseudo colors are drawn to group the regions that roughly share similar colors, as shown in 3(a). Only six strokes are drawn to label the image. Then, in each label region, we fine-tune the color by specifying colors for several pixels, as shown in 3(b). Only two pixels in each region are chosen to assign colors in this example. 3(c) The final colorization effect.

[58] introduced a colorization approach that utilizes examples and accounts for differences in lighting between the grayscale target image and the color source image. Initially, the method retrieves an illumination-independent intrinsic reflectance map of the target scene from various color references obtained through web searches. Subsequently, the target grayscale image is separated into its intrinsic reflectance and illumination components by utilizing the grayscale versions of the reference images. The color from the color reflectance map is then transferred to the grayscale reflectance image. Finally, the relighting process, which involves the target image's illumination component, produces the final colorization outcome. To implement this method, it is necessary to search for appropriate source images as references via web search.

The research conducted for the paper [12], introduces a colorization technique that involves a single parameter for grayscale images. The method computes the polynomial fitting model for the histograms of both the source image and the grayscale image using linear regression. By assigning an order to the polynomials as per the user's choice, the source image and the grayscale image can be automatically aligned. The colorization process is completed by transferring the color between the corresponding regions of the source image and the grayscale image.

3 Automatic Colorization

In contrast to semi-automatic colorization, automatic colorization requires no to minimal input from the user in the form of color seeds or scribbles to begin the colorization process. This was a more advanced and automated approach as compared to methods discussed in Semi-Automatic colorization. This method also becomes very useful when it is difficult to find correspondence between colors and texture between reference images and the image to be colored.

Li and Hao [11] discussed an automatic colorization approach using local linear embeddings. In this approach, given a color image of similar content as the grayscale image, the method couples them into similar patches which are distributed on a manifold [5] [2]. After this, for each patch the chromatic information is predicted/estimated by applying manifold learning using the neighbors in the training patches.

Chapiat el al. [3] in their work discuss a method in which they deal directly with multimodality and instead of choosing the most probable color at local level, they predict the probability distribution of all possible colors for each pixel. After this, they use graph cut technique to maximize the probability of the whole colored image globally.

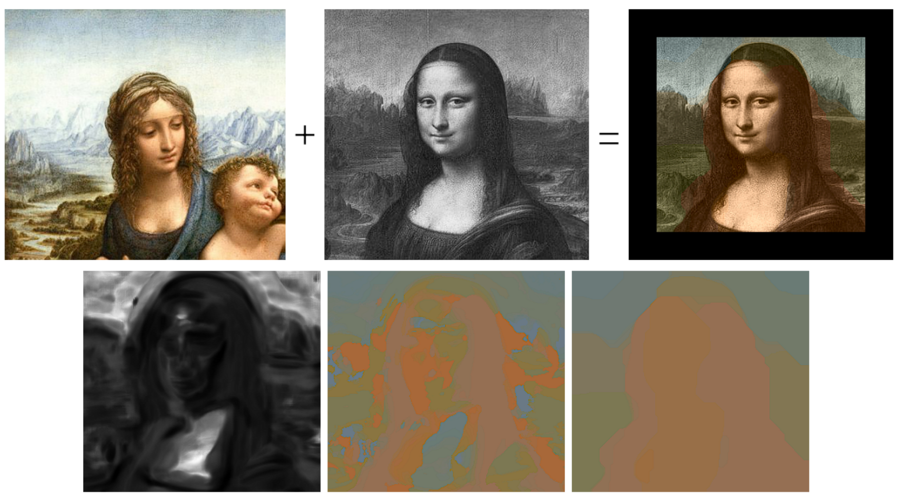


Figure 4: Automatic Colorization via Multimodal Predictions

In the above figure, the iconic Monalisa picture is colored using Madonna. In the second line, in the first picture, color variation is predicted where white stands for homogeneity and black for color edge. In the second picture, most probable color at the local level chosen by the algorithm is depicted and in the third picture the 2D color chosen by graph cuts are shown. In earlier methods, regions such as the neck or forehead would get colored blue as the skin looks like sky in those regions. However, in this method surroundings of these regions and lower probability colors play a decisive role in choosing colors finally for the grayscale image.

Liu and Zhang [13] presented an automatic grayscale colorization technique based on histogram regression. In this approach, the given source image and target grayscale image, locally weighted regression is performed on both the images to extract their respective feature distributions. These features are then matched automatically by aligning zero-points of the histogram. After acheiving the luminance-color correspondence, the grayscale image is colorized in a weighted manner.

This method gives considerable advancements and leaps in results as compared to previous methods but it may fail for images which have strong texture patterns or varied lighting effects such as shadows and highlights.

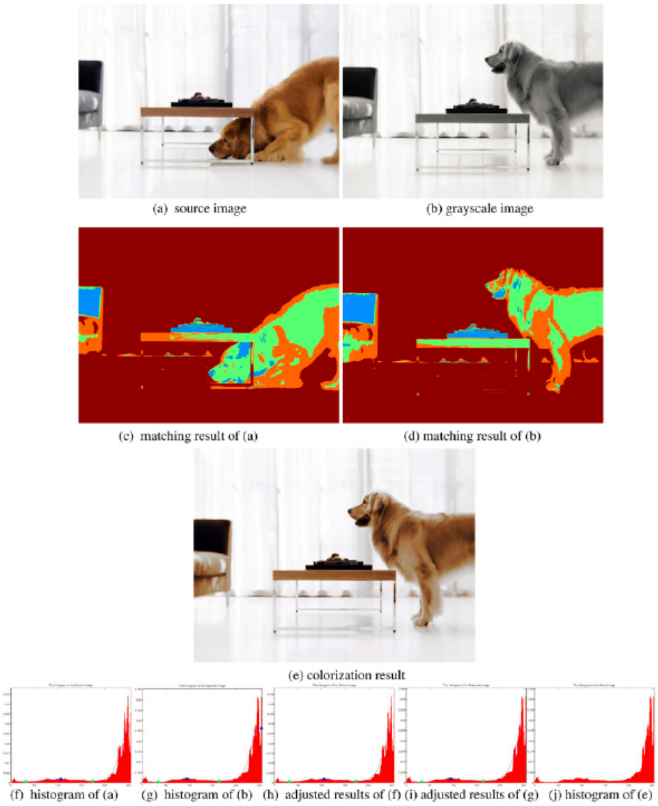


Figure 5: Automatic Colorization via Histogram Regression

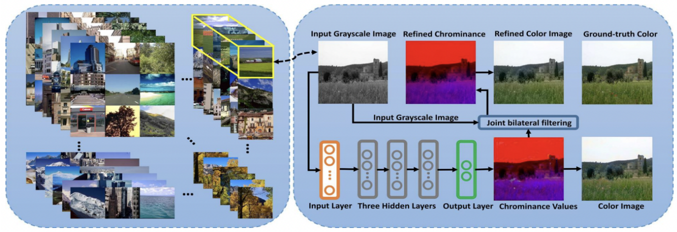
Liu and Zhang [14] proposed an enhancement of the technique discussed above which is based on texture map. In this method, given a source color image with similar content to the grayscale image, the method gathers luminance and texture information of images. In this, spatial maps of source image and grayscale image are computed where spatial map is a function of original image which indicates the luminance spatial distribution for each pixel. Then the process of locally weighted regression is used on the calculated spatial map and a series of spatial segments are computed. In each segment, the luminance of target grayscale image is mapped automatically to color values. Colorization results are then generated through local and global luminance-color correspondence between source color image and target grayscale image.

4 Deep Colorization

Deep colorization in computer vision is a method that involves using deep neural networks to automatically add color to black and white images or to enhance the color levels of existing-colored images. A large set of color images and related grayscale images are used to train a deep neural network so that it can recognize patterns and connections between them. A grayscale image is then taken as an input. The trained deep neural network is then used to predict the corresponding color values for each pixel in the input image. The network uses information from surrounding pixels, as well as global context information, to make its predictions.

Cheng et al's paper [4] implemented the method discussed above. The neural network was trained on a large set of source images from different categories with various objects.

In the paper [9], Larsson developed a deep learning-based approach for colorization by predicting per-pixel color histograms. Their method involves training a neural architecture in an end-to-end



(a) Reference image clusters (b) The proposed colorization method

Figure 6: Deep neural network for colorization

manner, considering semantically meaningful features of varying complexity. A color histogram prediction framework is then used to address uncertainties and ambiguities inherent in colorization. As illustrated in Figure 6, the mentioned approach involves using a deep convolutional architecture (VGG) to select spatially localized multilayer slices as per-pixel descriptors for a given grayscale image. Hue and chroma distributions are then estimated for each pixel using its hyper column descriptors. During testing, the estimated distributions are utilized for color assignment.

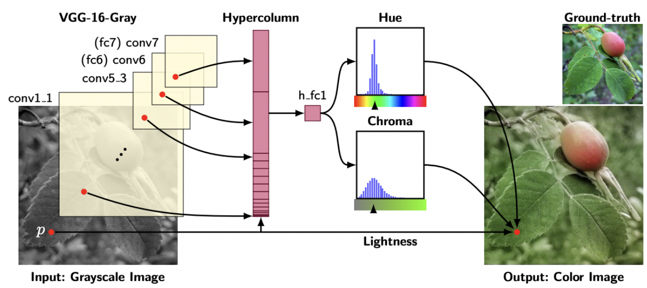


Figure 7: Larrson's system architecture overview

Iizuka in paper [7] proposed a CNN-based method consisting of 4 components, namely a low-level features network, a mid-level features network, a global features network, and a colorization network. This network architecture is able to jointly extract global and local features from an image and fuse them for colorization.

Zhang et al. [20] in their work define the colorization problem as a classification task taking into consideration the uncertainty associated with the colorization task. Their main contributions in this work are in designing an appropriate objective/loss function that handles multimodal uncertainty of the classification problem and captures a wide variety of colors, introduction of a novel framework for testing colorization algorithms and setting of a new high-water mark on the task by training on a million images.

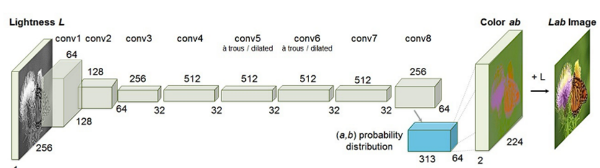


Figure 8: Network Architecture used in Zhang et al. 's work.

The CNN which they have used in their work is as above where each convolution layer refers to a block of 2 or 3 repeated conv and ReLU layers followed by a Batch Normalization layer. This has no pooling layers. The resolution changes are achieved using down sampling and up sampling between conv blocks.

In earlier work Euclidean loss is used as an objective function to minimize loss during training. The Euclidean Loss between predicted and ground truth colors is defined as:

$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\sum_{h,w}\|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

Here, H, W are image dimensions and predictions have a symbol 'hat' on top of the variable. The averaging effect favors grayish, desaturated results which deviate from the ground truth.

Instead, they have treated this problem as multimodal classification. They have quantized the output space by experimentation into bins with grid size 10 and keep Q as 313 which are in-gamut. They have used a multinomial cross entropy loss in their method which is defined as –

$$\mathrm{L}_{cl}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\sum_{h,w} v(\mathbf{Z}_{h,w}) \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

Here 'v' is a weighing term that is used to rebalance the loss based on color rarity.

They have trained their network on 1.3 Million images from the ImageNet training set, validated on the first 10k images in the ImageNet validation set, and tested on a separate 10k images in the validation set. Evaluating the quality of synthesized images using techniques such as RMS error on pixels often fail to capture visual realism. To address this, they have used three metrics for quality of image produced. The first metric used here is AMT (Amazon Mechanical Turk), in this human participants were shown a series of pair images where each pair consisted of a color photo next to a re-colorized version. Participants were asked to select the photo they believed was fake. Each experiment consisted of 10 trials each. They report that in 32% of trials the participants were successfully 'fooled' by their proposed method. The second metric used is for semantic interpretability. In this, they tested by feeding the color image results produced by their architecture to a VGG network that was trained to predict ImageNet classes from real color photos. Their classifier achieved 56% accuracy and after fine tuning it reached 63.5% accuracy. The third metric used is for Raw Accuracy (AuC). In this, they compute the percentage of predicted pixel colors within a thresholded L2 distance of the ground truth in ab color space. They then sweep across thresholds from 0 to 150 to produce a cumulative mass function and integrate the area under curve (AuC) and normalize. This metric measures raw prediction accuracy.

Results obtained when using this method on legacy black and white photos. The photos from Left to right are: photo by David Fleay of a Thylacine, now extinct, 1936: photo by Ansel Adams of Yosemite; amateur family photo from 1956; Migrant Mother by Dorothea Lange, 1936 .

Zhang et al. [21] proposed a CNN based framework aimed toward user guided image colorization. The system directly maps a grayscale image, along with sparse, local user "hints" to an output colorization with a Convolutional Neural Network (CNN). Rather than using hand-defined rules, the network propagates user edits by fusing low-level cues along with high-level semantic information, learned from large-scale data. This system is aimed novice users to enable them to create realistic colorizations. To guide the user towards efficient input selection, the system recommends likely colors based on the input image and current user inputs. The colorization is performed in a single feed-forward pass, enabling real-time use.

Their method applied to legacy black and white photographs is depicted above. The images are : Top left: The Tetons and Snake River, Ansel Adams, 1942; Bottom left: Photo by John Rooney of Muhammad Ali versus Sonny Liston, 1965; Right: V-J Day in Times Square, Alfred Eisenstaedt, 1945.

5 Conclusion

Our literature survey has provided a comprehensive overview of how colorization methodologies in computer vision have evolved over the years. We have reviewed various semi-automatic colorization techniques, including source-target, luminance-key, and optimization-based methods. These techniques have made colorization more accessible to non-experts by providing user-friendly interfaces for adding color to grayscale images. Furthermore, we have explored deep colorization techniques, which use deep learning methods to automatically generate plausible color images from grayscale images. These techniques have shown great promise in generating high-quality color images without any human intervention.

Although convincing results have been achieved by the current colorization methods, we think some challenges still remain. Most current colorization methods struggle to handle complex scenes with multiple objects, intricate textures and complex patterns. Current methods often require manual intervention to ensure high-quality results for these types of images. Another challenge that still remains is with regards to the available datasets. Deep colorization methods require large-scale datasets for training, which can be challenging to obtain and annotate. Moreover, these datasets need to be diverse enough to capture the variability in natural scenes, adding to the difficulty of dataset curation. The future for this domain however looks bright as the attention given by researchers to find smart solutions for this problem has drastically increased over the past 5 years.

References

[1] Arash Abadpour and Shohreh Kasaei. An efficient pca-based color transfer method. *Journal of Visual Communication and Image Representation*, 18(1):15–34, 2007.

[2] David Beymer and Tomaso Poggio. Image representations for visual learning. *Science*, 272(5270):1905–1909, 1996.

[3] Guillaume Charpiat, Matthias Hofmann, and Bernhard Schölkopf. Automatic image colorization via multimodal predictions. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III 10*, pages 126–139. Springer, 2008.

[4] Zezhou Cheng, Qingxiong Yang, and Bin Sheng. Deep colorization. In *Proceedings of the IEEE international conference on computer vision*, pages 415–423, 2015.

[5] Wei Fan and Dit-Yan Yeung. Image hallucination using neighbor embedding over visual primitive manifolds. In *2007 IEEE Conference on computer vision and pattern recognition*, pages 1–7. IEEE, 2007.

[6] Rafael C Gonzales and Paul Wintz. *Digital image processing*. Addison-Wesley Longman Publishing Co., Inc., 1987.

[7] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 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 (ToG)*, 35(4):1–11, 2016.

[8] Revital Ironi, Daniel Cohen-Or, and Dani Lischinski. Colorization by example. *Rendering techniques*, 29:201–210, 2005.

[9] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 577–593. Springer, 2016.

[10] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. In *ACM SIGGRAPH 2004 Papers*, pages 689–694. 2004.

[11] Jun Li, Pengwei Hao, and Chao Zhang. Transferring colours to grayscale images by locally linear embedding. In *BMVC*, pages 1–10, 2008.

[12] Shiguang Liu. Two decades of colorization and decolorization for images and videos. *arXiv preprint arXiv:2204.13322*, 2022.

[13] Shiguang Liu and Xiang Zhang. Automatic grayscale image colorization using histogram regression. *Pattern Recognition Letters*, 33(13):1673–1681, 2012.

[14] Shiguang Liu and Xiang Zhang. Image colorization based on texture map. *Journal of Electronic Imaging*, 22(1):013011–013011, 2013.

[15] Qing Luan, Fang Wen, Daniel Cohen-Or, Lin Liang, Ying-Qing Xu, and Heung-Yeung Shum. Natural image colorization. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 309–320, 2007.

[16] Dongdong Nie, Lizhuang Ma, Shuangjiu Xiao, and XueZhong Xiao. Grey-scale image colorization by local correlation based optimization algorithm. In *Visual Information and Information Systems: 8th International Conference, VISUAL 2005, Amsterdam, The Netherlands, July 5, 2005, Revised Selected Papers 8*, pages 13–23. Springer, 2006.

[17] Erik Reinhard, Michael Adhikhmin, Bruce Gooch, and Peter Shirley. Color transfer between images. *IEEE Computer graphics and applications*, 21(5):34–41, 2001.

[18] Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller. Transferring color to greyscale images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 277–280, 2002.

[19] Liron Yatziv and Guillermo Sapiro. Fast image and video colorization using chrominance blending. *IEEE transactions on image processing*, 15(5):1120–1129, 2006.

[20] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 649–666. Springer, 2016.

[21] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S Lin, Tianhe Yu, and Alexei A Efros. Real-time user-guided image colorization with learned deep priors. *arXiv preprint arXiv:1705.02999*, 2017.