STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LXVI**, Number 2, 2021
DOI: 10.24193/subbi.2021.2.03

A REVIEW AND ANALYSIS OF THE EXISTING LITERATURE ON MONOCHROMATIC PHOTOGRAPHY COLORIZATION USING DEEP LEARNING

ALEXANDRU MARIAN ADĂSCĂLIŢEI

Abstract. It is universally known that, through the process of colorization, one aims at converting a monochrome image into one of color, usually because it was taken by the limited technology of previous decades. Our work introduces the problem, summarizes the general deep learning solutions, and discusses the experimental results obtained from open-source repositories. Although the surveyed methods can be applied to other fields, solely the content of photography is being considered. Our contribution stands in the analysis of colorization in photography by examining used datasets and methodologies for evaluation, data processing activities, and the infrastructure demanded by these systems. We curated some of the most promising papers, published between 2016 and 2021, and centered our observations around software reliability, and key advancements in solutions employing Generative Adversarial Networks and Neural Networks.

1. Introduction

Photography colorization, in the context of this paper, represents the procedure of artificially reconstructing color information in a picture that has never been captured on a storage medium capable of recording color. In the absence of such research, we took the challenge of providing a comprehensive perspective on deep learning solutions. Approaches vary, with examples of discriminative networks [4, 52], generative networks [15, 53], and adversarial ones [1, 6]. In an area which have seen solutions as early as 1980, and modern ones began to appear around 2002, only the recent years brought methods to yield impressive results. A discussion was opened on 28 papers and 17 datasets, leading us through four main patterns, each with different models, computational demands, and time costs. Through patterns, a common set of

Received by the editors: TBA.

2010 *Mathematics Subject Classification.* 68T99.

1998 *CR Categories and Descriptors.* I.4.8 [**Image Processing and Computer Vision**]: Scene Annalysis – *Color*.

Key words and phrases. Photography Colorization, Deep Learning.

input values and processing steps are grouped under a common name, while the models encompass technically detailed networks. However, color reconstruction may yield structurally incoherent result, due to the lack of similar visual content in the training phase of the models. To better assess, experiments were conducted on a dataset of our own, managing to confirm a common trend regarding colorization performance.

On the one hand, our research methodology aimed to study the existing patterns, highlighting the main differences. On the other hand, we focused on the architecture, and the use of data. Aside from photography, domains such as communication protocols, medical imaging, and gaming could benefit from data compression, physiological highlights, and photo-realistic scene renderings, respectively. This paper contributes with a thorough analysis of the existing papers and datasets, process guided by the following research questions.

1.1. Aims and Research Questions.

While our primary concern was the best possible coverage of the literature, software reliability and open access to the source code shaped our approach to a good extent, making us ask the following initial research questions `RQ 1` to `4`.

- `RQ 1` As for now, is automatic colorization achievable without deep learning?
- `RQ 2` What solving patterns and deep learning models are usually employed?
- `RQ 3` How well would these models perform in professional applications?

The rest of the paper analyzes the work of the previous half of a decade, with a five section structure, having the context and relevance of colorization described in Section 2, patterns and models of learning in Section 3, literature result analysis in Section 4, and the conclusions summarized in Section 5.

2. Context and Relevance

In explaining how light is stored on a computer, this section builds an intuitive reasoning path to understand why mathematical reconstruction formulas can not infer color directly from the grayscale image.

2.1. Digital Representation.

When it comes to the pictures we store on our computers, they can be thought of as grids of numerical values, stacked upon each other. In this stack, each layer stores in those numeric values information regarding light absorption, by converting the electromagnetic wavelengths to a color standard computers can reproduce. The purpose of such structures is to mimic the eye's response to natural light but in the context of a screen. For example, transitioning from the wavelength measurements to the 1931

Commission Internationale de l'éclairage (CIE) standard [41] one uses three empirically determined weights, $\bar{x}_\lambda$, $\bar{y}_\lambda$, $\bar{z}_\lambda$, called tristimulus. As they only measure the color perception, historically defined by a group of human observers, we will use these values to determine the actual components of CIE, namely X, Y, and Z.

Color Space. The fundamental aspect of black and white photography, and the reason behind lacking mathematical formulas for color reconstruction `RQ1`, is that the technology capturing the visible light, either film rolls or digital sensors, only keeps the brightness, a weighted average of the mixed wavelengths we call colors. What was previously a rich source of information, now it became unable to tell what colors were in the scene. Using the Lab format, one may train various models to predict the chromaticity based on the luminance channel, pretending it was the black and white image, and then, using the discarded channels, measure the distance between the prediction and the reality. A candidate model makes the transition from black and white to color stacking two layers of chromaticity, which in the case of Lab one describes blue/yellow, and the other one orange/violet information.

2.2. **Progress and Relevance Over Time.** Colorization is a practice as old as photography itself, dating back to the eighteenth century, but recent technological breakthroughs endowed us with tools capable to supplement the monochrome record with a more appealing visual representation. Recording light on a physical medium had to overcome the technological limitations imposed by the early photographic materials which only captured shades of black and white. At the beginning of the nineteenth century, around the year 1930 [21], color photography gave people access to an unconscious understanding of the physical world through color. Nowadays, teams of artists such as Dynamichrome [11], are closing the gap, manually reconstructing black and white records. They are deciding how to approach the task, using their experience and intuition. Similarly, deep learning algorithms are calibrating their parameters until the model is behaving as intended.

Methods that predate the 1980s had more of a mechanical nature, using some of the early iterations of computers capable of graphical manipulation. Although little research has been made publicly available until 2002, Wilson Markle and Brian Hunt patented one of the earliest, if not the first attempt of movie colorization [34] using a computer, in 1988. For each scene of a black and white film pellicle, a color mask was applied to the frame. The adjacent frames were then similarly colored, taking the motion into consideration. Each of the areas indicated as having motion was processed by some adjacent pixels algorithm, while static areas were inheriting the previously applied colors.

3. Colorization Patterns and Learning Models

The following section is dedicated to answering RQ2 through a brief introduction into the patterns' general idea, and a discussion on the models' traits.

3.1. **Colorization Patterns.**

The source of data and the processing pattern can have a decisive performance impact on the model. Predicting color channels depends on the type of information sources one has at their disposal, whether it implies contextual hints, or large amounts of color images.

3.1.1. *Data-Driven Colorization.* Due to the fact that early algorithms heavily relied on human interventions, the work that followed completely removed preferences coming from the outside of the system. Su et al. [42] separated the image as a whole from the objects within the frame, colorized each patch using a network inherited from [52], and later fused the features while also avoiding the artifacts. Even if limitations may appear when the object instances are not well detected, it usually generates results with fine-tuned details without human interventions. Presumably, from the idea of finding the best local match, and balancing global coherence, improved approaches will be derived. All such approaches leverage large scale data and end-to-end training. Nevertheless, the decision of relying of fully autonomous processes was later reverted, and human preferences began to be taken into account under various forms that will be discussed in the following paragraphs.

3.1.2. *Human-in-the-Loop Colorization.* The shared knowledge of a community, historical documents, or reference images contain information that artists may access and use in their work, yet software methods are still unable to deal with such diversity and spread of information when transferring color. The following methods embrace the multi-modality of the problem, providing colorization results that differ when changes are iteratively introduced by a person. From such interactions, reinforcement learning may better predict what would be of interest for humans in color photographs. However, seldom is reinforcement learning present in the scene of colorization networks.

Based on Textual Descriptions. Notes were often placed on the back of legacy photography, and even nowadays, colorization associated with a language has rich sources of training data. Many social media platforms are improving their indexing systems based on the words and sentences associated with the visual content. Photography colorization based on captions conditions the nuances to fit color palettes associated with the present words, building on the idea that particular colors are associated with complex semantic concepts. One may imagine that a *cold evening* varies in nuances of blue, while the *golden* *hour* covers everything in warm colors. Regions that could not be matched from the text are then processed using a dominant color, such as *denim blue*.

Manjunatha et al. [33] concatenated units of text into every convolutional block of the baseline network - a fully convolutional neural network, obtaining a model that joins textual and visual feature maps at the cost of significant parameter demands. To address the issue of parameter efficiency, the authors employed a second approach to fuse the representations, using a feature-wise linear modulation - Perez et al. [37]. Training on the dataset presented in Lin et al. [31] yield unsatisfactory colorization due to image complexity and the set size limitation, although accounting for over $82 \cdot 10^3$ images. In these circumstances, the baseline network was pre-trained on ImageNet, then the two network variants presented in [33] were fine-tuned. The evaluation confirmed a better precision in the second model, although no significant differences were observed in both evaluation metrics or Turing tests. Both models performed well under caption changes, and they were able to change the colorization according to the updated sentences.

Image segmentation based on natural language expressions was approached by Hu et al. [17], then based on their framework Chen et al. [7] improved on features fusion, using a recurrent attentive module for deciding the number of text-to-image processing iterations. The framework matches image regions with the words describing them, employing the recurrent attentive fusion module that repeatedly reads the textual features maps until, through the attention mechanism, enough information was retrieved. A deconvolutional network later takes the fusion features map and up-scales it to the width and height of the final image, with a depth comprised of the number of classes resulted in segmentation and two chromaticity channels. Additionally, Chen et al. [7] introduced the first colorization results obtained on Oxford-102 Flowers dataset [36].

Bahng et al. [3] proposed two generative adversarial networks, one for text-to-palette generation, T, and another one, P, for palette-based colorization. The generator of T learns mappings between color palettes and sequences of words, while the discriminator distinguishes between real and fake palettes, using the Huber loss across the network. P operates on two sub-networks, a U-Net based colorization network, and a network that guides the colorization based on the color palette generated by the caption, whose output is passed to a Deep Convolutional Generative Adversarial Network (DCGAN) discriminator. Provided with rich textual resources, the model generates multiple color palettes, adapting to more than a couple of words, thus contrasting the limitations of small input volumes that previous work brought. Following this idea, a dataset of more than 10^3 mappings between sequences of words and palettes of five colors was introduced, and later applied on T's training.

The language itself makes a great difference in colorization, as English has eleven basic color categories, Russian twelve, and other languages might drastically differ, with the number of color terms reaching as low as three - white, dark, and red. Berlin and Kay theory addresses how various cultures share a basic understanding of color, even if they have various manifestations at the vocabulary level. Loreto et al. [32] presented a multi-agent simulation on how the use of a language influences color terms.

Based on Color Hints. Learning deep priors was not always the obvious path, and the early colorization approaches were envisioned to spread color strokes in correlation with the luminosity channel. Deep image priors represent a network's ability to obtain some knowledge about the world, and then use it in the actual task, where such knowledge comes in handy, and alone it is not enough to find the answer we seek. Data-driven processing turned the coin, making the process easier at the expense of user control. Might the best of both worlds be obtained, then a truly robust tool would be handed to the creatives ones, allowing for colorization preferences that would be difficult to include otherwise. In the following paragraph, we explain how one may combine user preferences and deep priors. Such user preferences come under the form of hues defined at a specific point on the digital canvas (tablet/monitor). Aside from the arbitrarily selected hues, the color hints get propagated through the network after specifying them on the user interface.

Zhang et al. [52] beautifully covered the specifics of this pattern, employing a CNN fusing low-level features extracted from clues with high-level semantic information. The main branch uses a U-Net architecture, which additionally absorbs the sparse color points through a Local Hints Network, L, and either the histograms or the average saturation levels using a Global Hints Network, G. Whenever the preference for a color is expressed on a drawing pad, a recommendation of nine colors is obtained through running a k-means clustering on G's final per-pixel distribution. In our experiments, this method ranked first when using their baseline model - the one with no hints provided. The user interface requires up to 8GB of RAM for the Docker image, but the experience is impressive. The codebase can be used without the graphical part, as the repository is very well documented and maintained. As an improvement, Xiao et al. [48] allowed for both global and local hints to be provided concurrently, in contrast to only one type of hint at a time as it was permitter in the previous approach [52].

Based on Reference Color Images. Transferring the chromaticity information from a semantically related color image to a target $\mathtt{T}$ monochromatic image is the main focus of this paradigm, and whether the user provides a reference $\mathtt{R}$ color image, or the system manages to retrieve the appropriate one, the idea is to allow for a multi-modal colorization, which neural networks prevent from happening using the dominant colors they have learned. One could imagine passing colors from cherry blossom to a black and white Californian coast image, obtaining synthetic, but artistic pink waves. Finding images with similar semantics and luminance as the input we want to process might prove as difficult as giving the right hints. Thus, in He et al. [16] an image query reaches to a gray-VGG-19 which in turn, based on its class, and the cosine similarity of the tuples $(\mathtt{R_i}, \mathtt{T})$ computed using the features $\mathtt{F}^5_{\mathtt{R_i},\mathtt{T}}$ and $\mathtt{F}^6_{\mathtt{T},\mathtt{R_i}}$ from network's last convolutional layer and first fully-connected layer, narrows down the top n images, generating a global ranking. Then, further pruning is realized using semantic and luminance similarities, which are denoted in Equation 1 as the sum's two terms.

$$\mathtt{score}(\mathtt{R_i}, \mathtt{T}) = \sum_{\mathtt{p}} (\mathtt{d}(\mathtt{F}^5_{\mathtt{T}}(\mathtt{p}), \mathtt{F}^5_{\mathtt{R_i}}(\mathtt{q})) + \beta \mathtt{d_H}(\mathtt{C_T}(\mathtt{p}), \mathtt{C}_{\mathtt{R_i}}(\mathtt{q}))) \tag{1}$$

Where $\mathtt{i} = \overline{\mathtt{0}, \mathtt{n}}$, β has been empirically set to $\mathtt{0.25}$, and $\mathtt{T}$ is our grayscale image, for each point p from F^5_T the nearest neighbor q from $F^5_{R_i}$ is assigned so that the pair minimizes the cosine distance. Then, $\mathtt{C_T}(\mathtt{p})$ maps each point from the feature map $\mathtt{F}^5_{\mathtt{T}}$ to a grid cell from a down-scaled $\mathtt{16} \times \mathtt{16}$ resolution T, which in turn gets used in $\mathtt{d_H}$ to compute the luminance similarity. The semantic similarity directly applies the cosine similarity represented by $\mathtt{d}(\mathtt{x}, \mathtt{y})$. After the local ranking is determined, the reference retrieval algorithm yields the top-1 reference image. The visual attribute correspondence technique used is known as Deep Image Analogy, which is explained at length in the work of Liao et al. [30].

A general downside of this pattern are the unrelated spots that should be dealt with, thus He et al. [16] employs an end-to-end colorization subnetwork that simultaneously learns color sample selection, color propagation, and dominant color prediction on two sub-branches, one for chrominance, and one for perceptual correlation. While the chrominance branch propagates color samples extracted from the reference to the entire image, the perceptual branch makes a prediction for areas left uncolored by the reference, purely based on dominant colors learned from the large-scale training set. This pattern was also used in the work of He et al. [16], and Xu et al. [49].

3.2. **Deep Learning Models.** One would have to create a list of initial study sources, therefore we provided in Table 1 our recommendation in terms papers that would represent a good read. The `28` papers influenced our opinion on the matter, and although we did not refer directly to all of them, the manner we grouped and filtered may represent a valuable source of information. In addition, it would be fair to say that one approach does not account for all our expectations, thus we focused on their strengths at the network level, making small remarks visible on the fourth column.

Architecture	Datasets	Metrics	Strengths	Related studies
Convolutional Neural Networks	[5], [9], [13], [39], [44], [47], [55]	LPIPS, PSNR, SSIM	produces excelent predictions for first time encountered parts of an image	[4], [16], [18], [26], [27], [33], [42], [48], [50],[51], [52]
Network Refinement	[5], [14], [22], [36], [39], [46]		optimizes on conservative predictions	[2], [7], [8], [10], [15], [38], [40]
Transformer	[39]			[25]
Generative Adversarial Networks	[3], [23], [28], [39], [43], [47], [54], [55]		less artifacts, better skin nuances, reduced blue bias for clothing	[1], [3], [6], [12], [19], [20], [29], [35], [45]

TABLE 1. Literature recommendations with the codebase freely available on GitHub.

3.2.1. *Convolutional Neural Networks.* This class of models is known for the heavy use in computer vision tasks. In the larger scheme of discriminating or generating numerical values, starting from a `2D` tensor representing the luminosity, and ending up with two chromaticity tensors, the network's layers, made out of convolutional kernels, are optimized and interconnected to improve the end result. When convolved with the input, these filters are generating the feature maps. In colorization, two important aspects must persist: the image ratio, which can be managed with padding, and that one should avoid image distortions, preferring a stride operation for pooling in the case of downsampling. The input image resolution ranges between 64 and 512 pixels, while some models have no restrictions on the input resolution, but they yield results within the previous boundaries. In the Lab format, the color values range between $-$`128` and `128`, and get later transformed so that they match the last layer activation (for example, for `tanh` would range between $-$`1` and `1`).

In general, the spatial information gets encoded, and lost, in exchange for learning more about the input image, procedure associated with an encoder. It is common to notice that additional features are added, fusing them into the output of the encoder, as they give us a stronger sense that we are in the possession of an improved solution. As an example, Baldassare et al. [4] used a pre-trained Inception-ResNet-v2 for features extraction alongside the encoder. Then, the model is upsampling the compact representation, using as much of the first layers as it needs to bring back spatial information. While different approaches leverage different parts of the network to their advantage, they have something in common in the way all approaches try to compress as many and insightful features together and to create the chromaticity channels out of them. In addition, hypercolumns are often used in this context (for example in Larsson et al. [27]), because the last layer gives information too coarse to precisely localize chromaticity descriptors in the pixels space, thus storing the activation values for a pixel increases prediction accuracy in the deeper layers.

Iizuka et al. [18] designed their approach based on Krizhevsky et al. [24], with four components: three networks thought for low, middle, and global features extraction, and a colorization network. A particularity of their work was that they allowed for input files of any resolution, global image priors, and colorization style transfer. When fusing global features with a purpose similar to that of priors into the local features, the environmental information influenced the colorization, avoiding, for example, green nuances for the water surface. The model was trained exclusively on 224×224 pixels images from [55], augmenting via cropping from an initial 256×256 pixels, and randomly flipping in the vertical orientation. According to the authors, results may be obtained on one of NVIDIA® Tesla® K80 GPU cores, with a batch size of 128, and 11 epochs (accounting for $2 \cdot 10^5$ iterations), in approximately 3 weeks time. As a comparison point, in the work of Baldassarre et al. [4] the same GPU unit completed the training stage in 23 hours, using a batch of 100, and $6 \cdot 10^4$ images, supporting the previous time estimation for training on the entire ImageNet dataset (which contains $14 \cdot 10^6$ pictures). For He at al. [16], training for 10 epochs, with a batch size of 256, took 2 days on eight Titan XP GPUs. Two days were also enough for Xiao et al. [48] to train their model using a batch size of 50 images, $4 \cdot 10^4$ iterations on NIVIDIA's GTX1080Ti GPU.

Larsson et al. [27] discarded the classification layer of a VGG-16 and transformed this fully convolutional network into a model in which each pixel had a probability distribution assigned over 313 ab pairs, a quantized color space that may vary in size from one implementation to another. While the idea was gaining traction due to existing progress documented in Zhang et al. [51], it later influenced the hint-based work of Zhang et al. [52], in which it was shaped into a pixel-level color recommendation. A VGG inspired network was also used in Zhang et al. [51], adding depth, dilated convolutions, and an improved loss function in the form of a classification loss to compare the probability distributions, while also making use of class rebalancing, without which desaturated colors would have dominated. A VGG-19 was used in both the similarity sub-network and the colorization one in the approach of He et al. [16]. Other networks were remarked, such as the GoogleNet, AlexNet, and Capsule Neural Networks, as well as those described in the work of Guadarrama et al. [15] and Zhao et al. [53] which use generative models, with a Pixel Convolutional Neural Network in the first approach, and a color distribution generator, coupled with a pixelated semantic generator in the latter.

3.2.2. *Generative Adversarial Networks.* Such networks, abbreviated GANs, share a fair amount of traits with the work presented in the previous subsections, consisting of two smaller networks. As the name denotes, the two networks compete, having a generator network produce images indistinguishable from ground truth, and a discriminator classify which pair of images contains the original color version. The training ends when the classification no longer distinguished between the two types of images, real, and colorized. The target is to avoid conservative predictions, and allow multiple colorization results by varying the noise, thus offering highly realistic results. Conditional GANs are most often employed, as the grayscale image represent part of the input and it could not be transformed into randomly generated noise as the traditional models would need. The generator takes the monochromatic image as a prior, and later allows for multi-modality through noise applied in the form of dropouts, or multi-layer noise coupled with multi-layer conditional information.

While the work of Nazeri et al. [35] had both the discriminator and the generator implemented after the U-Net architecture, the work of Cao et al. [6] envisioned an alternative to the encoder-decoder structure. One may image the encoder-decoder structure, where the middle part contains a U-Net architecture with skip connections between the layer $\mathtt{i}$ and $\mathtt{n} - \mathtt{i}$ to compensate for the bottleneck that prevents the low level information to reach the last layers. Such approaches tend to process the overall image information, which is suitable for transformations at the whole image scale, but in the case of colorization, it lacks local guidance. Nazeri et al. [35] embraced this method, and noticed, among other things, improved performance in the generator's encoder when leaky ReLU was applied. However, Cao et al. [6] preserves details at their location in space by using only convolutional layers in the generator. The noise gets attenuated when introduced early, hence it would be beneficial to introduce it in multiple layers. Complementing it, the multi-layer conditional information may be easily achieved, due to the fact that the network never used spatial transformations that would have complicated the process. Both [35] and [6] were inspired by the work of Isola et al. [19], given that the general idea of image-to-image translation has strong points that could be adapted from case to case.

An insightful read is the work of Antic et al. [1], called DeOldify. To the best of our knowledge, it remains the only competitive approach that was not associated with a research paper. Antic introduced a new breed of architecture, called NoGAN. Independently training the generator and the discriminator gives us most of the insights we need, then, GAN training addresses the issue of colorization realism. Shortening the GAN's training manages to avoid artifacts formation, while also closing the gap towards vivid colors. When the two networks are to be trained together, an inflection point in training will be noticed shortly, marking the moment when the critic managed to reach a learning threshold. When reached, the training must end, otherwise the quality varies drastically. Although not yet defined, the inflection point was determined by saving the checkpoints at each 0.1% of training data, and then manually inspecting whether the quality of the images did abruptly drop. This approach offers an artistic model, addressing details and color saturation, and a stable one, tailored for landscapes and portraits. In the same category of rarely visited ideas, we noticed the PatchGAN discriminator employed in the work of Victoria et al. [45]. Further exploration regarding pixel-level independence between two patches could offer an excellent penalty system in colorization.

4. Literature Results Analysis

Since the early '80s, the number of solutions proposed in literature remained small, in the two digits figure, and out of those, the human eye may be fooled by only a dozen of these algorithms. To further support research initiatives in legacy photography colorization, we have manually curated a 102-photograph dataset, shot on both film and digital mediums. Table 2 presents the results obtained from a variety of techniques, studying the context in which these models perform best, but also when they reach their limitations. For example, we often encountered models poorly selecting color distributions for landscape scenes, while at the same time, accurate color palettes for portraits. The results presented in this table were obtained from the open-source implementation made available by the authors of these papers on GitHub. The initial codebase was not changed in any manner.

In Table 2, the three columns denoting metrics, `LPIPS`, `PSNR` and `SSIM` rank the models by statistical means, and they will be introduced in Section 4.2. A number of factors contribute to a low score, such as patches left untouched, colors mappings without any real grounds, or spots leaking color into the immediate vicinity. Most models process landscapes and nature scenes well, while only particular portraits, urban events, and outdoor activities may deceive a person. Even if the work of Antic et al. [1] and Iizuka et al. [18] sometimes yields an unconvincing version of reality, it is impressive how those colors can, at the same time, provide a starting point for artists, and a bridge to the past for the general public. The last column summarizes the type of images we believe, based on the experiments, that would optimally be colorized. We aligned our results with those obtained in He et al. [16], Su et al. [42], and Zhang et al. [52], thereby agreeing with the general trend.

An improved performance can be observed on the generative models' side. The first column ranks the performance starting from the lowest score, while the other two columns rank in the opposite order. The metrics may have specific ranges of values, yet it remains a problem specific issue. The colorization has, as for the moment, no testing methodology, and this state of development leaves an opportunity for further research initiatives. To answer `RQ3`, the existing methods can deliver when used in professional photography tasks, being integrated into products targeting the general public. One example is the work of Zhang et al. [52] that was included in Photoshop Elements 2020.

Paper	Colorization Metrics						Recommended
	↓ LPIPS	σ	↑ PSNR	σ	↑ SSIM	σ	types of images
Zhang et al. [52]	0.11678	0.04927	18.69112	3.41512	0.88102	0.08394	all
Iizuka et al. [18]	0.18068	0.06863	15.80264	3.94617	0.77813	0.12155	events, portraits, landscapes
Antic et al. [1]	0.18389	0.08614	13.36557	3.55204	0.73828	0.12560	all
Zhang et al. [51]	0.22174	0.08790	13.60779	4.01649	0.77388	0.11998	landscapes
Kumar et al. [25]	0.30766	0.07357	11.22693	3.14602	0.53996	0.15731	close-up portraits, landscapes

TABLE 2. Performance evaluation made on a 102-image dataset (`github.com/alexdarie/color/images`) containing urban landscapes and events, objects, and portraits.

4.1. **Datasets Challenges.** The main disadvantage when solving this task is the training data, as we encountered only a hand full of datasets specifically designed for the task, as for example the Palette-and-Text dataset [3], or the Chinese Youth Subculture dataset [29]. Aside from these, the existing solutions inherited the most popular computer vision training sources. An overview can be found in Table 1. Often, images from other tasks are either semantically too simple, too small resolution-wise, or they lack descriptors (textual or color clues), thereby partially preventing the learning process. Although they might seem numerous, the existing sets lack diversity present in consistent amounts. Such a balanced dataset would take some of the time spent on adapting to data, and move it towards learning from it.

4.2. **Evaluation Metrics.** Three metrics are most often used to assess the results, namely Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and the Learned Perceptual Image Patch Similarity (LPIPS), yet they might neglect the human intuition with respect to the goal. Notable about the first two would be that the PSNR centers around the MSE, while SSIM is defined using three factors: luminance, contrast, and structural similarity. In the case of LPIPS, it learns the similarity using deep neural network activation function values.

An alternative to these metrics, recently highlighted, is the use of the Patch-based Contrast Quality Index (PCQI), and the Underwater Image Quality Measure (UIQM). Nevertheless, when the human intuition is the next in line, our recommendation is to have a prior empirical study, and an open mind, as they are designed to address colorization efficiency, and not data compression loss. PCQI accounts for the mean luminosity, change in contrast, and structural distortion, while UIQM requires no reference image, and measures sharpness, colorfulness, and contrast.

Despite all the effort, having a person assessing the colorization results remains the golden standard at the moment, as mathematical observations may miss important aspects. A test involves a number of correspondents answering whether they think that the photography they see was colorized or is the original one. Out of the total amount of trials, a fooling rate is determined, accompanied by the probability that an observation occurred by chance.

5. Conclusions and future work

The work presented in this paper sets the grounds for further colorization initiatives. We initially explored whether data driven colorization may achieve human level accuracy, and discovered that there are cases when it is possible. Even alone, the fact that colorization optimizes time costs, and reduces manual labor allows the general public to relive moments from their collection of old photographs. Moreover, the tasks deriving from colorization have even wider implications. Even if this challenge is governed by the absence of a dedicated dataset, and the tendency to borrow techniques from image compression, the generative models, and even the more straight forward convolutional neural network can achieve impressive results.

The gap formed by the semantically complex images, will, in time, be closed through optimizations specific to computational photography. The work of Antic et al. [1], and Zhang et al. [52] would be our recommendation as a model development gateway. As for solving the open problems, enough room was left for improvement in areas such as color leaks, color normalization, conservative predictions, as well as the resolution constraints. Making the colorization models more accessible to the general public, and improving on the existing approaches are the milestones we set for ourselves in the future.

Acknowledgments

The project development relied on the High Performance Computing Center from the Babeş-Bolyai University, Cluj-Napoca, Romania.

References

[1] Antic, J. jantic/deoldify: A deep learning based project for colorizing and restoring old images (and video!). `github.com/jantic/DeOldify`, accessed on Dec 4, 2020.
[2] Ardizzone, L., Lüth, C., Kruse, J., Rother, C., and Köthe, U. Guided image generation with conditional invertible neural networks, 2019.
[3] Bahng, H., Yoo, S., Cho, W., Park, D. K., Wu, Z., Ma, X., and Choo, J. Coloring with words: Guiding image colorization through text-based palette generation, 2018.
[4] Baldassarre, F., Morín, D. G., and Rodés-Guirao, L. Deep koalarization: Image colorization using cnns and inception-resnet-v2, 2017.
[5] Caesar, H., Uijlings, J., and Ferrari, V. Coco-stuff: Thing and stuff classes in context, 2018.
[6] Cao, Y., Zhou, Z., Zhang, W., and Yu, Y. Unsupervised diverse colorization via generative adversarial networks, 2017.
[7] Chen, J., Shen, Y., Gao, J., Liu, J., and Liu, X. Language-based image editing with recurrent attentive models, 2018.
[8] Cheng, Z., Yang, Q., and Sheng, B. Deep colorization, 2016.
[9] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding, 2016.
[10] Deshpande, A., Lu, J., Yeh, M.-C., Chong, M. J., and Forsyth, D. Learning diverse image colorization. `github.com/aditya12agd5/divcolor`, 2017.
[11] Dynamichrome. Showcase. `dynamichrome.com`, accessed on Dec 4, 2020.
[12] El Helou, M., and Süsstrunk, S. BIGPrior: Towards decoupling learned prior hallucination and data fidelity in image restoration. *arXiv preprint arXiv:2011.01406* (2020).
[13] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, 2012.
[14] Gross, R., Matthews, I., Cohn, J., Kanade, T., and Baker, S. Multi-pie. In *2008 8th IEEE International Conference on Automatic Face Gesture Recognition* (2008), pp. 1–8.
[15] Guadarrama, S., Dahl, R., Bieber, D., Norouzi, M., Shlens, J., and Murphy, K. Pixcolor: Pixel recursive colorization, 2017.
[16] He, M., Chen, D., Liao, J., Sander, P. V., and Yuan, L. Deep exemplar-based colorization, 2018.
[17] Hu, R., Rohrbach, M., and Darrell, T. Segmentation from natural language expressions, 2016.

[18] Iizuka, S., Simo-Serra, E., and 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* (07 2016), 1–11.

[19] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks, 2018.

[20] Kiani, L., Saeed, M., and Nezamabadi-pour, H. Image colorization using generative adversarial networks and transfer learning. In *2020 International Conference on Machine Vision and Image Processing (MVIP)* (2020), pp. 1–6.

[21] Kodak. Chronology of film. www.kodak.com/en/motion/page/chronology-of-film.

[22] Krizhevsky, A. Learning multiple layers of features from tiny images. Tech. rep., 2009.

[23] Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 dataset. https://www.cs.toronto.edu/~kriz/cifar.html.

[24] Krizhevsky, A., Sutskever, I., and Hinton, G. Imagenet classification with deep convolutional neural networks. *Neural Information Processing Systems 25* (01 2012).

[25] Kumar, M., Weissenborn, D., and Kalchbrenner, N. Colorization transformer. github.com/google-research/google-research/tree/master/coltran, 2021.

[26] Larsson, G., Maire, M., and Shakhnarovich, G. Colorization as a proxy task for visual understanding. github.com/gustavla/self-supervision, 2017.

[27] Larsson, G., Maire, M., and Shakhnarovich, G. Learning representations for automatic colorization. github.com/gustavla/autocolorize, 2017.

[28] Li, Y., Zhuo, J., Fan, L., and Wang, H. J. Cys: Chinese youth subculture dataset. https://github.com/tezignlab/subculture-colorization/tree/main/CYS_dataset.

[29] Li, Y., Zhuo, J., Fan, L., and Wang, H. J. Culture-inspired multi-modal color palette generation and colorization: A chinese youth subculture case, 2021.

[30] Liao, J., Yao, Y., Yuan, L., Hua, G., and Kang, S. B. Visual attribute transfer through deep image analogy, 2017.

[31] Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. Microsoft coco: Common objects in context, 2015.

[32] Loreto, V., Mukherjee, A., and Tria, F. On the origin of the hierarchy of color names. *Proceedings of the National Academy of Sciences of the United States of America 109* (04 2012), 6819–24.

[33] Manjunatha, V., Iyyer, M., Boyd-Graber, J., and Davis, L. Learning to color from language. github.com/superhans/colorfromlanguage, 2018.

[34] Markle, W., and Hunt, B. Coloring black and white signal using motion detection. *Canadian Patent Nr 1291260* (01 1988).

[35] Nazeri, K., Ng, E., and Ebrahimi, M. Image colorization using generative adversarial networks. *Lecture Notes in Computer Science* (2018), 85–94.

[36] Nilsback, M.-E., and Zisserman, A. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing* (Dec 2008).

[37] Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A. Film: Visual reasoning with a general conditioning layer, 2017.

[38] Royer, A., Kolesnikov, A., and Lampert, C. H. Probabilistic image colorization. github.com/ameroyer/PIC, 2017.

[39] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. Imagenet large scale visual recognition challenge, 2015.

[40] Santhanam, V., Morariu, V. I., and Davis, L. S. Generalized deep image to image regression. `github.com/venkai/RBDN`, 2016.

[41] Schanda, J. *CIE 1931 and 1964 Standard Colorimetric Observers: History, Data, and Recent Assessments*. Springer New York, New York, NY, 2016, pp. 125–129.

[42] Su, J.-W., Chu, H.-K., and Huang, J.-B. Instance-aware image colorization. `github.com/ericsujw/InstColorization`, 2020.

[43] Timofte, R., Agustsson, E., Gu, S., Wu, J., Ignatov, A., and Gool, L. V. Div2k dataset: Diverse 2k resolution high quality images.

[44] Tyleček, R., and Šára, R. Spatial pattern templates for recognition of objects with regular structure. In *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 364–374.

[45] Vitoria, P., Raad, L., and Ballester, C. Chromagan: Adversarial picture colorization with semantic class distribution, 2020.

[46] Wang, L., Guo, S., Huang, W., Xiong, Y., and Qiao, Y. Knowledge guided disambiguation for large-scale scene classification with multi-resolution cnns. *IEEE Transactions on Image Processing 26*, 4 (Apr 2017), 2055–2068.

[47] Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2010), pp. 3485–3492.

[48] Xiao, Y., Zhou, P., and Zheng, Y. Interactive deep colorization with simultaneous global and local inputs, 2018.

[49] Xu, Z., Wang, T., Fang, F., Sheng, Y., and Zhang, G. Stylization-based architecture for fast deep exemplar colorization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 9360–9369.

[50] Yang, Z., Liu, H., and Cai, D. On the diversity of realistic image synthesis. `github.com/ZJULearning/diverse_image_synthesis`, 2017.

[51] Zhang, R., Isola, P., and Efros, A. A. Colorful image colorization, 2016.

[52] Zhang, R., Zhu, J.-Y., Isola, P., Geng, X., Lin, A. S., Yu, T., and Efros, A. A. Real-time user-guided image colorization with learned deep priors, 2017.

[53] Zhao, J., Han, J., Shao, L., and Snoek, C. G. M. Pixelated semantic colorization, 2019.

[54] Zhou, B., Khosla, A., Lapedriza, A., Torralba, A., and Oliva, A. Places: An image database for deep scene understanding, 2016.

[55] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. Learning deep features for scene recognition using places database. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1* (Cambridge, MA, USA, 2014), NIPS'14, MIT Press, p. 487–495.

Department of Computer Science,, Faculty of Mathematics and Computer Science,, Babeş-Bolyai University, Kogălniceanu no. 1

Email address: `aaic2261@scs.ubbcluj.ro`