Journal of University of Babylon for Pure and Applied Sciences, Vol. (28), No. (1): 2020

Developing Agent-Based Model for Colorization

Ahmed A. Husssein Mahdi Abed Salman Hasanain Ali Al Essa
Nawras Yahya Hussein

University of Babylon, Iraq
mahdi.salman@uobabylon.edu.iq

ARTICLE INFO
Submission date: 18/ 9 / 2019
Acceptance date: 8 /12 /2019
Publication date: 31/ 3 / 2020

Abstract

Colorization is adding colors to black and white images. Colorization attracts the interest of researchers as it serves wide are of applications such as web technology and medical images processing. The expected large dimensionality of source datasets, imposes the automation is mandatory. In this paper, a natural inspired solution is developed for automatic color transferring from colored to grayscale images. The proposed algorithm can be easily implemented in parallel and distributed environment. We show that our technique can be applied on broad image types, with preserving image features such as texture and luminance. The resulting images make our technique applicable in verity domains. The algorithm is simulated and result is presented using NetLogo tool.

Keywords: colorization, automatic color transfer, agent-based modeling.

(a) (b) (c)

Figure 1: Colorization example: (a) The source color image, (b) the target grayscale image and (c) image colorized by the proposed method.

1. Introduction

The dramatic development of information technology, especially web technology, in recent decades attracts more attention for reproducing old-grayscale images as colored images through what known as image colorization. Adding colors to black and white photo is as old as photo itself and its challenge problem, tedious and requires advanced labor-intensive [1].

Broadly, colorization algorithms can be divided into three main groups: manual, Interactive and automatic algorithms. In manual, the user colorizes each region of segmented image by hand [2]. In interactive algorithms [3-5], the user suggests seed colors for each region of the target image, and the colors propagate to whole region automatically with respect the boundaries. Finally, automatic algorithms [6-8], transfer colors from a colored to grayscale images without user intervention.

The idea of color transfer is taking a colored image as a reference and transfer colors to the target grayscale image. The result is the colorized image visually appealing and perceptually meaningful.

A key challenge of colorization problem is replacing each pixel of one dimension vector from grayscale image with three dimensional vector of reference color image (e.g. blue, yellow and green). Hence, there is an urgent need for human intervention in this process.

Colorization process has a multi of practical applications: such as computer games, colorizing old films or photos, correcting color in old photographs and converting the amounts of comics from black and white into colorized versions. Furthermore, the problem is a good model for a wide range of problems. One advantage of colorization as a model is that immense colorization datasets are easily available, and they are organized in interesting ways [9].

This paper presents a distributed method of transferring colors from colored to grayscale images automatically without any intervention from the user of course with an exception of choosing the source image.

Natural societies of insects inspire researchers for developing distributed algorithms such as Ant colony [10] and Bees colony[11] which is applied broadly for optimizing many solutions. To the best of our knowledge, we the first use Bee Colonies algorithms to improve the solution for image colorizing. Such algorithms can be applied in a distributed or parallel computing environment. In other words, the improving is getting exponentially.

The autonomous of the process and the quality of the results obtained over a set of several selected images demonstrate the efficiency of our method.

Fig. 1 shows an example of colorize a grayscale image, given a source colored image (left) and the target grayscale image (middle). Our method uses Bees to choose similar color pixels from the source image and transferring colors to the output image, the method produces colored images (right), with no human intervention.

This paper is organized as follows: Related works are reviewed in section two. Section three presents the methodology of the research. Results are presented and discussed in section four. Finally the study is concluded in section five.

2. Related Works

In recent decades, there have been many advances in the field of colorization that propagate colors supported by the user as a hints to the whole image or transferring colors to grayscale images.

Hwang et. al. [12], solve the problem by taking a statistical-learning-driven approach and they build and design a convolutional neural network to convert a grayscale image to colorize output. The output depends on the images learned in the past step by the proposed system.

Richard Zhang et. al. [13], design an objective function that processes the multi-modal uncertainty of the colorization problem and catches a wide variety of colors. They introduce a modern framework for evaluating colorization algorithms, where the colorization problem is introduced as a straightforward and competitive method for self-supervised learning, obtaining satisfactory results on many criteria.

Kevin Frans [14], uses two distinct convolutional networks in tandem for obtaining colored and shaded artwork from outlines. The problem breaks down into two stages. First, the color prediction network infers a general color scheme from an outline. Then, a final image is produced by taking this color from the shading network.

Richard Zhang et. al. [15], use a deep learning approach for colorization process. Along with sparse, the proposed method directly matches a grayscale image with the scribbles color by the user to capture the result with a Convolutional Neural Network. By fusing low-level cues along with high-level semantic information, the network propagates the color hints introduced by the user. For real-time use, the colorization is performed in a single feed-forward pass.

Satoshi Iizuka et. al. [16], also use Convolutional Neural Networks, but unlike others, the method for fusing the local and global features lets model to be run on any resolutions of input images. The proposed method extracts low-level features of the image and then these features used to compute global and mid-level image features. After these features are fused, the method produced the chrominance map.

Aditya Deshpande et. al. [9], propose a method learns from a reference color image to produce a colorize result. In the chromaticity maps the method trains a quadratic objective function, similar to a Gaussian random field. On the image features, the coefficients of the objective function depend and images are then colorized by minimizing this objective function.

Benoit Arbelot et. al. [17], define regions of images as areas of similar textural content by finding textural information while preserving edges. To transfer color between regions, proposed method computed on a large scale regions to be able to find the properties of the large textures. The work proposes an edge-preserving texture based on region contrast, letting color transformations.

Kazunori Uruma et. al. [18], propose a new method for image coding that need a few representative pixel (RPs) and gives precise information about chrominance. Their method consists of two phases, coding phase (extraction RPs) and decoding phase (colorization). The method extracts RPs from a variety of multi-resolution images gained by variety of down-sampling and error minimizing of colorization. In the stage of colorization, the algorithm colorizes multi-resolution images.

3. Research Methodology

The proposed method is implemented as Agent Based Model ABM. An agent is a human or machine that can interact with other agents and environment. The agent is characterized by its ability to take decisions autonomously [19]. Taken a decision is translated as an action that may change self-state or/and the state of the environment. The agent behaves according to some designation rules. Finding a good rule play a key role in the modelling process. In ABM the behavior of agent is encoded an a method inside an encapsulated object that also contains references for representing the internal state. The agent should be able sense (read) the state of the environment. Agent is an implementation for artificial ants or bees. Note in this context, the word bee or agent is used interchangeably.

The mission of our bees is colorizing each pixel of a grayscale image by searching and copying color information from a region of chosen color image.

The texture of both (source and target) images need to be computed on same color space for comparison. The color space and the transformation is presented in section 3.1.

3.1 Transferring Colors

In this study, YUV color space [20] is extracted from RGB space where the luminance of the pixels is represented by the vector Y in a grayscale image. While the chrominance information represent by the vectors U and V which are provided by the reference image.

Two images (the reference and grayscale image) are adjusted in one new image to create a representative field for bees (virtual field) that has half-colored and half-gray scaled pixels as shown in Fig. 2. Following [6], for each pixel of the merged imaged, the standard deviation (texture measure) is computed on Y vector for (5x5) neighbor pixels centered in the given pixel.

The parameters of the algorithm are adjusted by setting the number of bees and the threshold of the similarity. The number of bees depends on the size of the image. The threshold of the similarity is determined, based on the accuracy of the colorizing and desired speed. The higher of the similarities, the faster of the transferring colors will be obtained.

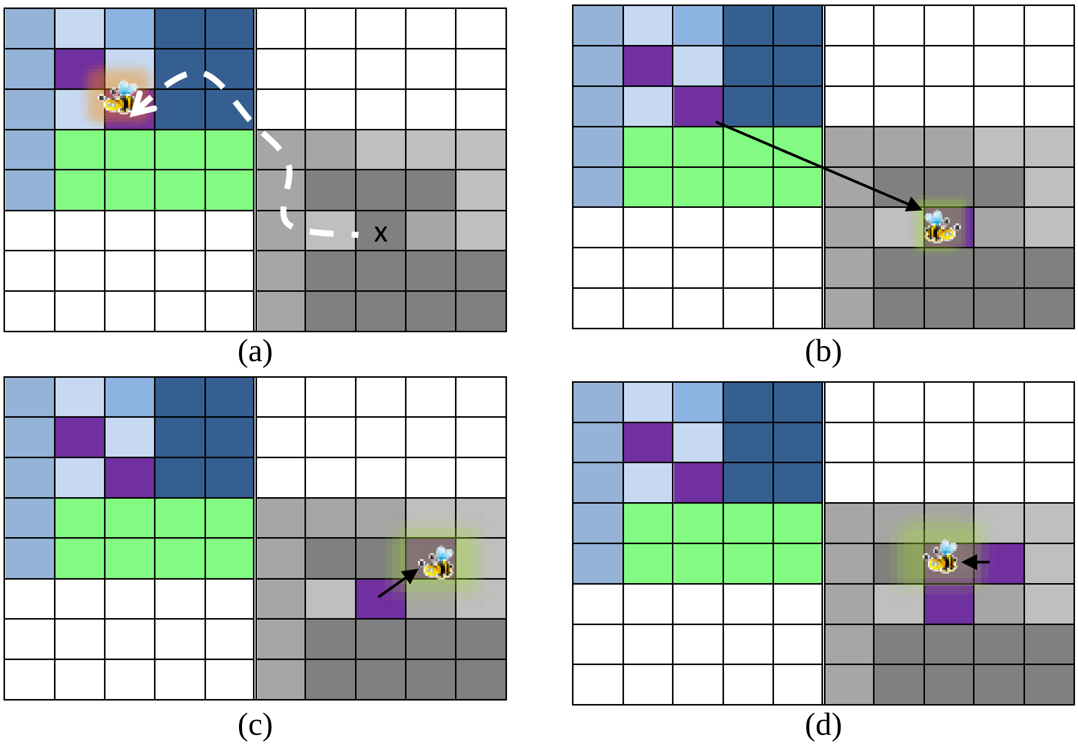


Figure.2 A bee in different state: (a) looking for color (b) coloring target pixel (c) and (d) coloring neighbor pixels.

3.2 Agent behavior

An agent behaves according to its state and the state of the environment. An agent has three states: {find-gray-pixel, find colored-pixel, painting}. States are coded as {0, 1, 2} respectively. The pseudo-code for agent behavior on each state is shown in Algorithm 1(a,b,c).

Algorithm 1: Agent behavior

(a) Find_gray_pixel

```
If(isGray(current_pixel) and (not reserved(current_pixel))
    Set px = pixel.x
    Set py = pixel.y
    Set Pstd = pixel. Similarity
    Set state = 1
    setReceived( current_pixel)
else
    move_to_a_neighbor_pixel
end
```

(b) Find_colored_pixel

```
If(isColored(current_pixel) and (isSimilarto(pixel(px,py)))
    Pick_color()
    Move_to(px,py)
    Drop_color()
    Set State = 2
else
    move_to_a_neighbor_pixel
```

end

(c) Painting

Let P = set of neighbor similar gray pixels
If P $\neq \emptyset$
    Let p $\in$ P
    move_to p
    drop_color()
else
  set state = 0
end

A bee in state 0 follows algorithm 1c. Bee looks for a gray pixel that is not reserved by another bee for coloring. If such pixel is found, its information such as coordinates and similarity measure are kept in mind. Then the pixel is marked as reserve red and the agent changes its state to 1. Otherwise, the bee moves to a neighbor pixel at random.

When a bee becomes in state 1, it looks for a colored pixel that has similar texture. It compares the similarity measure in hand with that of each colored pixel in the way. Passing by such pixel, the bee picks color information. Then, it goes back to the grayscale pixel. It drops the color information i.e. changing the pixel from YUV to RGB. The bee changes its state to 2. Otherwise, the bee moves to neighbor pixel at random. See algorithm 1b.

A bee in state 2 is painting the neighbor grayscale pixels that are match to similarity measure in mind following algorithm 1c. It keeps painting whenever there is such pixels in the neighborhood. If bee does not find any more pixels of same texture, it resets its state to 0.

4. Results and discussion

We tested our method on selected images from Free Big Pictures dataset [21] to measure our system's ability to produce colored images. Free Big Pictures is a collection of free and high resolution images contains several thousand images partitioned into categories.

For all our experiments, a fixed number of bees are used since images are taken same in size. However, the number of bees affects only the speed of the process which is not considered in this study. The similarity threshold is taken 0.2 between the source and target pixels.

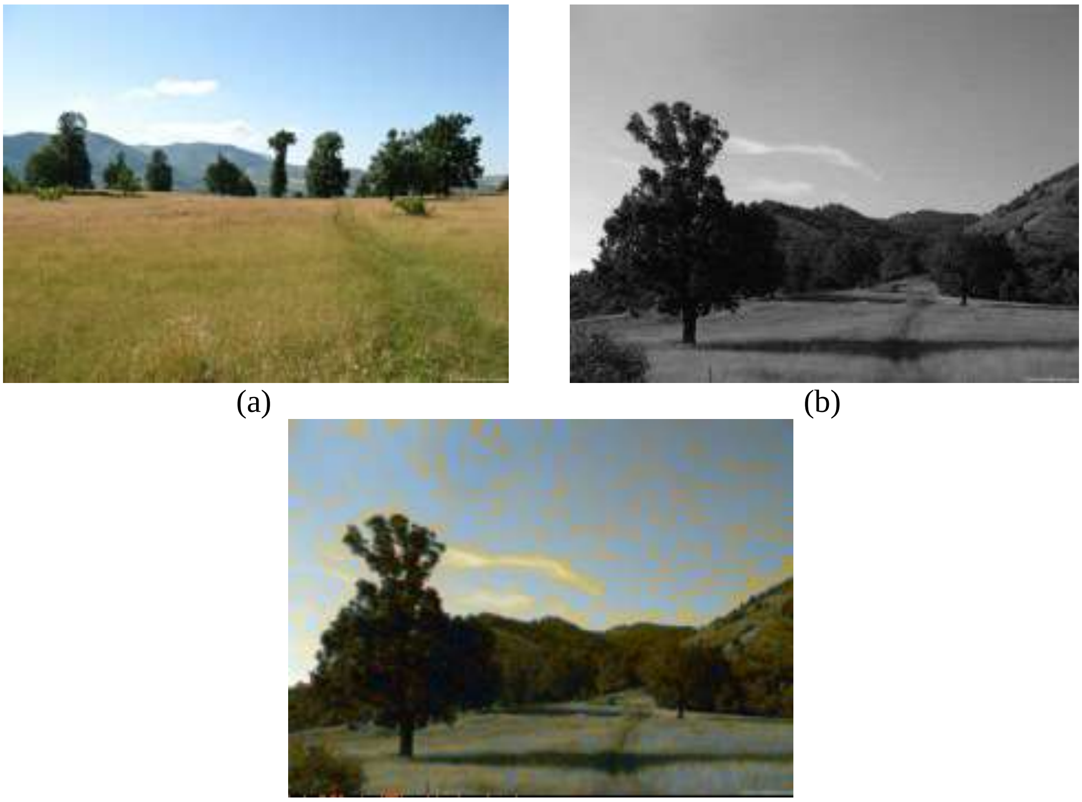
Many experiments have been conducted on verity types of images. All experiments produce relatively same output quality. Hence, three experiments chosen arbitrary are presented here. All image are taken of (172 × 230) pixels in size. They are shown in figures Fig. 3, fig 4, and fig 5 show the source image in (a), (b) for the grayscale image, and the resulted image in (c).

In all resulting images, experiment show the colorization has reasonable saturation and the colors are transfer to restricted regions that are corresponding in texture.

For example, the trees in foreground of image are colorized with high saturation of green with keeping the fine details with the sky in the foreground of image.

However, we notice in some regions there is a noticeable amount of noise. This results of using simple similarity measure even though, it can be improved using a more robustness one.

Output shows that our method produces a lively, nicely results with managing most of the details such grass, trees, mountain and the sky. Overall, the colorized results has highly appeal than the input images with keeping the accurate details of the boundaries of the objects.



(a) (b)

(c)

Figure 3 experiment on images with different details types (sky, tree, grass, etc.): (a) source image (b) grayscale image (c) output image.

(a)

(b)

(c)

Figure 4: Experiment on images of trees: (a) source image (b) grayscale image and (c) the result

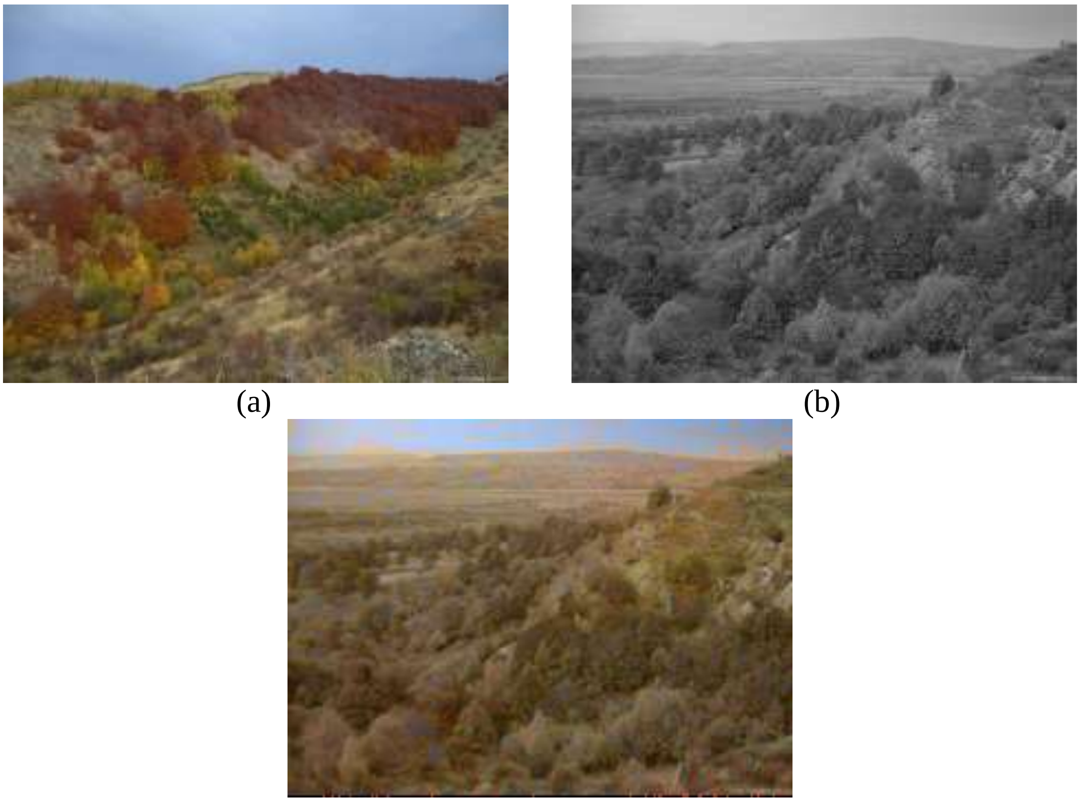

(a) (b) (c)

Figure. 5 Experiment on images of grass: (a) source image (b) grayscale image and the result images in (c).

Fig. 6 shows the distribution of bees on missions over time. The yellow curve represents the number of bees engaged in searching for colored pixels. The red curve refers to the number of bees that are looking for non-colored pixel. Finally, the green curve is the number of bees that are diffusing colors in their neighbors on non-colored pixel that are similar to the current pixel. It's clear that the number of bees engaged in the mission of finding similar colored pixel is high. The reason is using of simple similarity and non-guided search which may accelerate the searching process.

Fig. 7 shows the number of pixels that have not been colored yet. At the beginning, many pixels are colorized in short time and the curve is approaching fast to zero. The reason for this is that each new colored pixel becomes a near source for coloring its region letting bees carry colors from local pixels. Then, it takes long time to colorizing a few number of pixels. For some individual pixels that are spread over the image, bees may take long time searching for similar colored pixels.

Finally, our method relies on the availability of color exemplar which is semantically similar to the gray image.

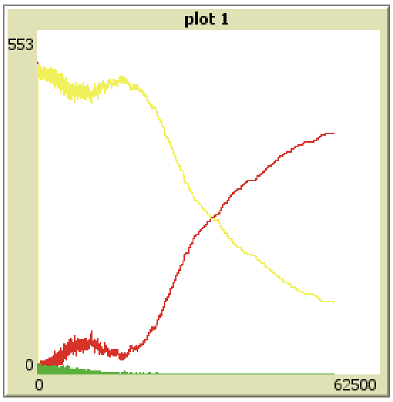


Figure 6: distribution of bees on each mission over time

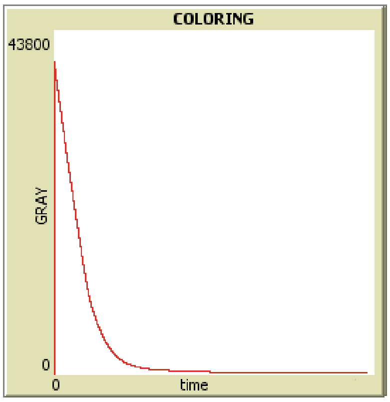


Figure 7: colorization process convergence

5. Conclusion

In this paper, a natural inspired solution for colorization is presented. An agent-based model is developed for transferring colors automatically between colored and grayscale images. The primarily results are considered promising foundations for developing more reliable distributed model suitable for parallel and distributed environments. The process is completely automatic. However, the source image should be chosen to have most color and texture that the destination may have. Developing an automatic selection approach of source image makes the model convenient for offline processing on clusters or cloud. In the future work the behavior of bees would be optimized in addition to use another similarity measure in order to reduce the amount of noise.

Conflict of Interests.
There are non-conflicts of interest

6. References

[1] D. L. a. Y. W. Anat Levin, "Colorization using optimization," ACM Trans. Graphics, vol. 23, no. 3, p. 689–694, 2004.

[2] " Art & Architecture Thesaurus, s.v. "hand coloring"," Accessed 22 November 2010..

[3] X. Y. Ahmed A. Hussein, "Colorization using edge-preserving smoothing filter," Signal, Image and Video Processing, vol. 8, no. 8, p. 1681–1689, 2014.

[4] S. H. K. T. M. L. Minh Ha Quang, "Image and video colorization using vector-valued reproducing Kernel Hilbert spaces," Journal of Mathematical Imaging and Vision, vol. 37, no. 1, p. 49–65, 2010.

[5] M. S. D. a. G. D. FINLAYSON, "Improvement of colorization realism via the structure tensor," Int. J. Image Graphics, vol. 11, no. 4, p. 589–609, 2011.

[6] M. A. K. M. Tomihisa Welsh, "Transferring color to grayscale images," ACM Trans. Graphics, vol. 21, no. 3, p. 277–280, 2002.

[7] M. H. B. S. Guillaume Charpiat, "Automatic image colorization via multimodal predictions," in In: Proceedings 10th European Conference Computer Vision, Marseille, France, 2008.

[8] S. D. N. B. H. B. Kekre, "Colorization of grayscale images using Kekre's biorthogonal color spaces and Kekre's fast codebook generation," Adv. Multimed. Int. J., vol. 1, no. 3, p. 49–58, 2011.
[9] J. R. R. D. A. F. Aditya Deshpande, "Learning large-scale automatic image colorization," in 15th IEEE International Conference on Computer Vision, ICCV, Santiago, Chile, 2015.
[10] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," Theoretical Computer Science, vol. 344, pp. 243 - 278, 2005.
[11] P. M. M. E. P. D. L. A. Yuce B, "Honey Bees Inspired Optimization Method: The Bees Algorithm," Insects, vol. 4, no. 4, pp. 646-662, 2013.
[12] K. M. R. T. Tung Nguyen, "Image Colorization with Deep Convolutional Neural Networks," in Proc. of ASIAGRAPH, Toyama, Japan, 2016.
[13] P. I. a. A. A. E. Richard Zhang, "Colorful Image Colorization," in ECCV, 2016.
[14] K. Frans, "Outline Colorization through Tandem Adversarial Networks," arXiv:1704.08834, 2017.
[15] J.-Y. Z. P. I. X. G. A. S. L. T. Y. a. A. A. E. Richard Zhang, "Real-time user-guided image colorization with learned deep priors," ACM Transactions on Graphics (TOG), vol. 9, no. 4, 2017.
[16] E. S.-S. a. H. I. Satoshi Iizuka, " 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, vol. 35, no. 4, p. 110:1–110:11, (2016.
[17] B. a. V. R. a. H. T. a. T. J. Arbelot, "Automatic Texture Guided Color Transfer and Colorization," in Expresive '16 Proceedings of the Joint Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering, Lisbon, Portugal, 2016.
[18] K. K. T. T. T. F. Kazunori Uruma, "Image colorization algorithm using series approximated sparse function," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 2014.
[19] J. Ferber, Multi-agent systems: An introduction to distributed, First Edition ed., Boston, MA: Addison-Wesley Longman Publishing Co, 1999.
[20] K. Jack, Video Demystified: A Handbook for the Digital Engineer(3rd), Ch.3, LLH Technology Publishing, , Eagle Rock, 2001.
[21] "Free Big Pictures," 2009-2016. [Online]. Available: http://freebigpictures.com/. [Accessed 27 Nov. 2017].

الخلاصة

التلوين هو اضافة الالوان الى صور الابيض والاسود لتصبح صور ملونة. التلوين يجذب الباحثين لكونه يخدم كثير من التطبيقات مثل صفحات الويب ومعالجة الصور الطبية. ان حجم البينات المتوقع يفرض ان تكون العملية الية. في هذا البحث، تم تطوير طريقة مستوحاة من الانظمة الطبيعية لنقل الالوان من الصور الملونة الى الصور الرمادية. يمكن تنفيذ الخوارزمية المقترحة بسهولة في انظمة الحوسبة المتوازية او الموزعة. كما يمكن تطبيق النموذج على انواع متنوعة من الصور مع الحفاظ على خصائص الصور كالاضاءة والنسج. اظهرت الصور الناتجة امكانية تطبيق الاسلوب في مجالات متنوعة. تم اجراء محاكاة لطريقة الحل باستخدام بيئة النت لوكو.