ColorNet: Deep Convolutional Neural Network for Grayscale Image Colorization(2018)

Áron Ritter, *BSC* *Student, BME,* Zoltán Máté, *MSC* *Student, BME*, and Dóra Istenes*, Bsc Student*, BME

**Abstract— The colorization of black and white images using neural networks is a relevant topic in the 21th century. The task meets all requirements of a Deep Learning task. We focused on adhering to this idea and prepared a learning network aimed to provide a solution.**

**Our project consists of two parts. The first part is the collection and preprocessing of the data, making it the most ideal for the network to use. Carrying out the automation of the data collection and the transformation on the dataset by using complex resizing and saving functions has proven to be a great task. The second part of our project and the main character of this article is the network. In spite of trial and error, heaps of failures we are of the opinion that our network is on the right track of solving this exciting and quite challenging problem. Furthermore, we implemented one deep convolutional network with two custom loss functions which we were inclined to compare. Due to the convolutional network we could have used images of any size but we settled with a resolution of 128\*128 in the name simplicity**

**Absztrakció-**

*.* **A fekete-fehér képek neurális hálózatokkal való színezése napjainkban igencsak aktuális téma.**

**A probléma teljesíti egy DeepLearning feladat elvárásait. Ebből kiindulva, a feladatra való megoldást kínáló tanulási hálózatot készítettünk. A projekt két részből áll. Elsőként az adatok összegyűjtése és előfeldolgozása került megvalósításra, ami a hálózatunk számára a leginkább ideális. Ezen belül az adatgyűjtés és átalakítás automatizálásának elvégzése az adatkészletben, összetett átméretező és mentő függvények használatával komoly feladatnak bizonyult. A projektünk második része amely egyúttal cikkünknek a lényege: maga a hálózat. Sok próbálkozás és hiba ellenére azon a véleményen vagyunk, hogy hálózatunk fejlesztése jó úton halad ennek a kihívást jelentő, ámde felettébb izgalmas probléma megoldásában. . *Furthermore, we implemented one deep convolutional network with two custom loss functions which we were inclined to compare.* A konvolúciós hálózatnak köszönhetően tetszőleges méretű képet is fel tudna dolgozni a hálózat, de 128 \* 128 felbontással dolgoztunk az egyszerűség kedvéért**

# INTRODUCTION

With the ever-improving deep learning, many problems, which were thought to be unsolvable for computers without human interactions, became feasible. One such problem is colorization of grayscale images, which dilemma is easily solvable for human imagination. Even if we cannot restore the ground truth, we can make plausible guesses, with our prior knowledge (e.g. the grass is green, the sky is blue etc.). The problem is straightforward. When an image is converted to grayscale two out of three dimensions is lost to us. So we have a strongly under determined problem. Where our network task is to learn the semantics priors to produce a colorized picture, which is plausible for the human eye.

Today the colorization is completed by humans who need to do comprehensive research if they want to uncover the ground truth. Colorization methods, which assist in their endeavor can be divided into two categories: scribble-based colorization [http://www.mirlab.org/conference\_papers/international\_conference/ACM%202005/docs/mm351.pdf , <http://webee.technion.ac.il/people/anat.levin/papers/colorization-siggraph04.pdf>, <http://www.cse.cuhk.edu.hk/~ttwong/papers/manga/manga.pdf> ], example-based colorization [https://people.cs.clemson.edu/~jzwang/ustc13/mm2012/p369-gupta.pdf, <http://www.cs.tau.ac.il/~dcor/online_papers/papers/colorization05.pdf> ]. The scribble-based requires significant effort from the user, where the example-based expect similar reference images, which is why it is greatly reduces the usability and increases the time needed to colorize a picture.

Our fully – automatic method is presented to tackle this problem. For vision problems the widely accepted network is Convolutional Neural Network. Which was used highly effectively to tackle many similar problems.

# Related works

. This section gives a brief overview of the previous colorization methods.

## Scribble-based colorization:

Requires the user to manually specify the colors of the regions. These scribbles color information is then propagated to the rest of the target image under the assumption that adjacent pixels with similar luminance has similar color. More advances exploit edges to reduce color bleeding.

## Example-based colorization

## Unlike the previously mentioned method example-based colorization transfer the color information from one or more reference images. The mapping between the images is established manually and/or using semblance between local descriptors.

The reference images can be provided from the user or the method can utilize the massive image data on the internet, in which case the user had to provide

## Automatic colorization

There are methods for automatic colorization:

<https://www.lri.fr/~gcharpia/colorisation.pdf>

<https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Deshpande_Learning_Large-Scale_Automatic_ICCV_2015_paper.pdf>

<https://arxiv.org/pdf/1611.07004.pdf>

<https://arxiv.org/pdf/1603.08511.pdf>

<http://iizuka.cs.tsukuba.ac.jp/projects/colorization/data/colorization_sig2016.pdf>

unit tesla). Refer to “(1),” not “Eq. (1)” or “equation (1),” except at the beginning of a sentence: “Equation (1) is ... .”

# Network architecture and training

A képen LEGO látható

A leírás nagyon megbízhatóWe got our motivation from [http://iizuka.cs.tsukuba.ac.jp/projects/colorization/data/colorization\_sig2016.pdf ], where they used a fully convolutional network in accordance with a classification network (between the two network the lower layers where shared). Their solution had the advantage of being able to recognize local and global priors from the images, as a result to their approach they were able colorize most of the input to plausible color. To simplify their network, we omitted the classification. And introduced concatenate layers as it can be seen on [figx], with this approach, which were founded in [: <https://arxiv.org/pdf/1505.04597.pdf>], we made a U-net architecture, instead of maxpooling layers we used convolutional layers with stride of two.

The network consists of 13 2D convolutional layer, where all of their activation was ReLu, except the last one, which used Sigmoid activation. All used (3, 3) kernel and the stride was (1,1) except where stated otherwise. After each convolutional network we used batch normalization [ https://arxiv.org/abs/1502.03167 ] The first 5 layers of our network was joined to their mirror part concatenate layer. (1 conv – 5 concatenate etc.). We used zero padding so the convolutional layers input and output where the same, except when the conv. layers stride is 2. After each halving we doubled the filter numbers. And after each upsampling, the conv. layers halved the filter numbers. the next 2 conv. layers (11, 12) are there for bigger feature recognition. Our last conv. layer with filter of 2 is the output

# Data Collection

In comparison to the network, the preparation and preprocessing of the data seems small, bit it is an underrated task. Without a big enough database, or one that doesn’t contain fitting images for the task, can derail the training in various ways, like overfitting or -in case of a too small dataset- it can’t learn enough to complete it’s task.

## The Dataset

First, we had our eyes on the ImageNet dataset. But after not being granted access to the original pictures, we searched for another option.

After careful consideration we decided on the following database containing 100000 pictures, in a 10G zip file, publicly available. <https://datasets.figure-eight.com/figure_eight_datasets/open-images/test_challenge.zip>

## The Algorithm

In order to not to burden the user, with having to download and extract the dataset, themselves, we made an algorithm to take care of the job. For a better insight, it tracks its progress, by timestamping the exact moment it starts a new process.

# data transformation

## Size and scale

Due to the learning algorithm being a convolutional network, we could have worked we images of any size. But taken into consideration our limited resources we deiced to side with a resolution of 128\*128. In order to not damage the layout of the picture, we cut them to a ratio of 1:1. We accomplished this by

calculating the shortest distance to the edge from the center and cut the longer side to the same measurement. For the downscaling, we used the resize function from the PIL.Image python library. We specified the LANCZOS resampling to achieve better quality in the transformed dataset.

## Saves

In order to achieve the possibly best outcome in training, we used high quality in the saving algorithm. The better quality of the transformed to pictures helped the the algorithm better in the learning process

## Color Scales

We used an LAB color scheme for the project. The LAB contains a grayscale channel, which makes the images easier to convert to grayscale, as we only had to remove the other 2 channels. Also the network only has to predict 2 channels instead of three.

# possible improvements

As of now we can start to notice that our database is to small for such a broad topic, one improvement would be to use a bigger database. Also, we it is possible to test more loss function and other hyperparameter for better results.

# Conclusion

Our network is learning, the implausible colorizations where attributed to the small database.

For every 100 batch (~1340 batch/epoch) we predicted images and saved them. We can observe that the colors are ever changing, most likely because of the batches containing more in one color than in the others. This is in place with what we learned

# References

http://www.mirlab.org/conference\_papers/international\_conference/ACM%202005/docs/mm351.pdf

<http://webee.technion.ac.il/people/anat.levin/papers/colorization-siggraph04.pdf>

http://www.cse.cuhk.edu.hk/~ttwong/papers/manga/manga.pdf

https://people.cs.clemson.edu/~jzwang/ustc13/mm2012/p369-gupta.pdf]

http://www.cs.tau.ac.il/~dcor/online\_papers/papers/colorization05.pdf

<https://www.lri.fr/~gcharpia/colorisation.pdf>

https://www.cv-foundation.org/openaccess/content\_iccv\_2015/papers/Deshpande\_Learning\_Large-Scale\_Automatic\_ICCV\_2015\_paper.pdf

<https://arxiv.org/pdf/1611.07004.pdf>

https://arxiv.org/pdf/1603.08511.pdf

[http://iizuka.cs.tsukuba.ac.jp/projects/colorization/data/colorization\_sig2016.pdf, ketszer kell hivatkozni ra]

http://iizuka.cs.tsukuba.ac.jp/projects/colorization/data/colorization\_sig2016.pdf

: <https://arxiv.org/pdf/1505.04597.pdf>

https://arxiv.org/abs/1502.03167

