Night Vision Image Colorization

Han Chenfeng and Ishita Gupta

Abstract

Our project proposes two methods for colorizing night vision images to compare the accuracy of object detection. After our experimental analysis, we found that colorization did not bring much change to the accuracy of object detection. We give analysis conclusions based on different methods. In the first method, because we used the L value of the night vision image, the L value of the night vision image is black or dark in a large range, and because the L value has not been processed, our colorization effect is not ideal. In the second method we deal with this problem and give acceptable colorization effects. However, because the resolution of the image itself is too low, the effect of colorization is not ideal, too. Based on our current experiments, we find that colorized images do not give significant improvements in object detection results. We give the future development direction of this project, which is to improve the resolution of the image to increase the coloring effect to a greater extent.

Introduction

In the field of computer vision, many object detection functions are based on color images, that is, daytime images. These images are the same as the scenery seen by the human eye. And use these images to complete the function of object detection through deep learning. Because the color differences of these images are obvious, and the colors of objects and backgrounds usually have a strong contrast, the neural network can learn colors as features when learning the object detection of these images. However, not all images are suitable for learning color as a feature. Examples are black and white images, grayscale images, and night vision images. We are interested in how object detection will behave based on these colorless image, and we are also interested in studying whether colorizing a colorless image can improve the accuracy of object detection.

We use night vision images as our research direction. Night vision images are derived from road monitoring at night, or photos taken at night with a night vision device. For security purposes, these images may be used for object detection. So it is important to improve the accuracy of night vision images. The night vision image is different from the grayscale image. In night vision images, most of the colors are black or dark, so the image looks dark overall. The grayscale image looks evenly black and white. An example of a noticeable difference is that the sky in grayscale or black and white images is white, while the sky in night vision images is black.

We use the colorization network to colorize the night vision image. Then compare the object detection accuracy of the colorized image with the night vision image. We used several different methods to achieve this process. In the process, we also reflected on whether colorization has a significant impact on the accuracy of object detection. This report will introduce the methods and results of our experiments in detail.

Related work

<u>Colorization</u>

Zhang et al. (2016) presented a new approach of colorization. They pose the colorization problem as a classification task and use class-rebalancing at training time to increase the diversity of colors in the result. The system is implemented as a feed-forward pass in a CNN at test time and is trained on over a million color images (Zhang et al. ,2016). With this approach, they give significant improvement on colorization and successfully fools humans on 32% of the trials on a“ colorization Turing test” which is significantly higher than previous methods. They also show that colorization can be a powerful pretext task for self-supervised feature learning, acting as a cross-channel encoder. This approach results in state-of-the-art performance on several feature learning benchmarks (Zhang et al. ,2016).

Object detection

Currently, object detection algorithms are mainly divided into two categories. The first one is based on regional proposals, such as R-CNN, Fast R-CNN and Faster R-CNN. The second one is based on end-to-end learning, such as YOLO and SSD.

The R-CNN algorithm mainly includes regional proposal, normalization processing, feature extraction, classification and regression (Girshick et. al., 2016). Based on R-CNN, Fast-RCNN optimizes the whole network by adaptive scale pooling, which avoids the redundant feature extraction operation in R-CNN and improves the accuracy and speed of network recognition (Girshick, 2015). For Faster R-CNN, it extracts the candidate frames by constructing the Region Proposal Network (RPN) instead of the selective search method with large time overhead, which further improves the speed (Ren et. al., 2015).

For the other stream, YOLO combines object detection and recognition, which greatly improves the speed (Redmon et. al., 2015). Its background false detection rate is lower than R-CNN and supports the detection of unnatural images (Redmon et. al., 2015). However, one disadvantage is that the object positioning error can be large due to the rough division of the S×S grid (Redmon et. al., 2015). The SSD improves the idea of the regional proposal and uses an RPN network similar to that of Faster R-CNN (Liu et. al., 2016). The difference is that the SSD uses RPN on multiple feature layers of CNN for classification and Bounding-Box regression (Liu et. al., 2016). Therefore, the detection of small objects on the image becomes more accurate.

Methods

Here we propose two different methods for Night Vision Image colorization. Our first method is based on the original colorization technique proposed by Zhang et al. However through our experiments we concluded that our initial method can not provide satisfactory results due to flaws. Hence we propose a second method for colorization of night vision images, which can provide competent results and help in increasing object detection in few scenarios.

For the color space, we use the CIELAB color space. It expresses color as three values: L* for the lightness from black (0) to white (100), a* from green (−) to red (+), and b* from blue (−) to yellow (+).

Method 1 - colorization using Lab color space

training phase :

1. We obtain a black-white night vision and colored image pair of the same scenario
2. Calculate the Lab metrics for both images in the pair
3. Use the L value from black and white night vision images as training input for CNN architecture
4. Use the ab values from colored image to calculate perceptual loss and update weights to repeat training process

Evaluation:

1. Calculate the Lab coefficients of a black-white night vision image as input value
2. Use the trained network to predict the ab metrics from the calculated L value
3. Concatenate the predicted ab values with the input L to obtain the Lab values for the output colorized image

4. Convert the Lab values to RGB and display image
5. Use pre trained network for object detection on the predicted image

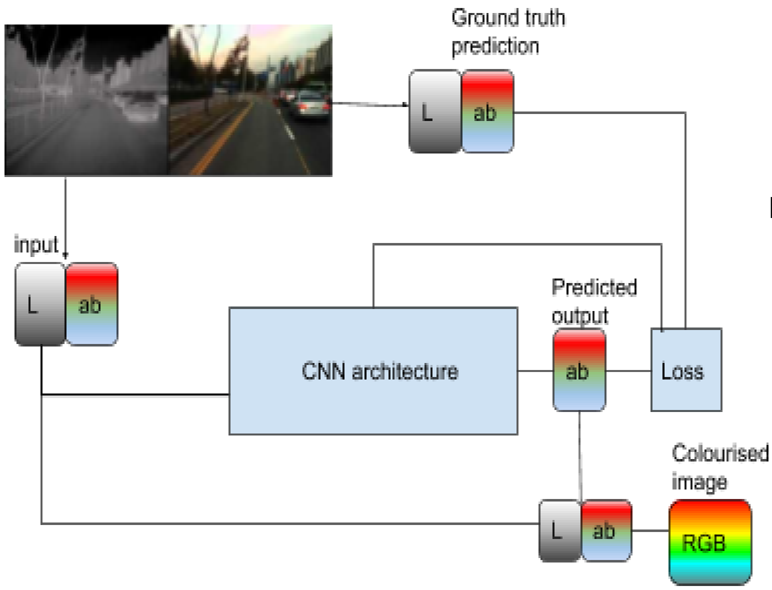


Fig. Night vision image colorization method 1 training proposal

However our method 1 did not yield acceptable results. We will analyse the reasons for this failure in detail with examples in the next section. Our next proposed method was based on the idea that we can exploit the preexisting large dataset of black-white images and their colored images. As night vision images have low resolution and no large dataset, training a neural architecture on it does not give strong results. Hence we decided to preprocess the nightvision images to obtain inverted images closer to their black-white counterparts and use a pretrained network to obtain stronger results.

<u>**Method 2- Inverted colorization**</u>

training phase :

1. We obtain a black-white night vision and colored image pair of the same scenario
2. Process the night vision image using the python imaging library inversion tool to obtain an image (negative-positive image inversion)
3. Use the obtained inverted image as input for a pretrained network architecture trained on large dataset
(The rest of the process is the same as the method 1 except with L values obtained from the processed image at step 2)

Evaluation:

1. Obtain colorized image from training phase
2. Perform object detection using powerful object detector such as Google Vision

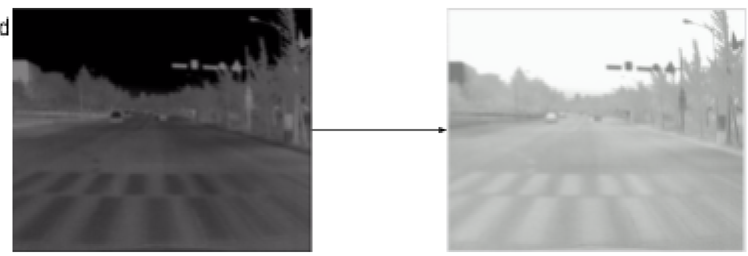


Fig. Preprocessing the night vision image using image inversion

Experiment

Dataset

For this project all the experiments were conducted using the KAIST Multispectral Pedestrian Dataset consisting of 95k color-night vision pairs (640x480). The dataset was prepared using a color camera, a night vision camera and a beam splitter to capture the aligned multispectral (RGB color + night vision) images. As we needed a dataset consisting of night vision (black-white night vision) images and their corresponding colored images, this dataset is ideal for that purpose. However due to computation restrictions we decided to use only 250 color-night vision pairs for our training set and 50 images for our validation set.

Network Architecture

Baseline model: We used a pre-trained black-white image colorization network to obtain colorized results of a night vision image. As our technique was based on modifying the technique used by Zhang et al in Colorful Image Colorization. We used their publicly available model trained on the ImageNet dataset.

Model 1 (colorization using Lab color space): we built our own network from scratch based on the paper by Zhang et. al for Image colorization. Following a similar architecture we have a encoder-decoder structure of convolutional layers with relu activation for each layer. The original paper used a modified loss function to support the prediction of rare colors in increase the vibrancy of images. However as our dataset consisted mostly of pictures of streets and roads we didn't have that requirement. Our loss function was mean squared error and Adam optimiser. Our code for this experiment can be linked from our colab file.

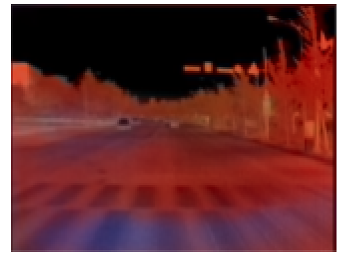
Model 2 (Inverted colorization): we used a NoGAN network to train image to image. For this part of the experiment we used a pretrained network based on the MIT repository DeOldify . However we added preprocessing code using the Python Imaging Library to modify it for our night vision images. We also used the Google Vision API for object detection on the obtained images.

Experiment results

For this project we mostly only performed qualitative tests on a few samples of our dataset to check the colorization output. As our main objective was to provide a new technique for night vision image colorization and explore the object detection accuracy on such colorized images vs. night vision images.

Baseline results -

Input Image



Output Image

Model 1 -

Input Image

Output colourised Image

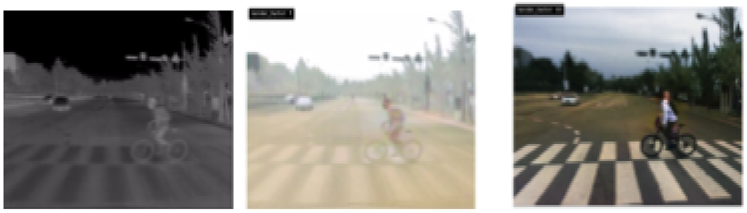
Failure Analysis:

Our baseline results are not accurate as predicted since the network is not trained to colorize night vision images. Hence there is a large amount of error in the prediction of colors and the scene is not colorized correctly. However even our method 1 results are not satisfactory as we do not see any kind of colorization in the output image except a slight yellow-ish tint. This is despite having a large training and validation accuracy (>80%) in predicting the correct ab coefficients of the colorized image.

The lack of color is due to the large L coefficient values of the Input image being concatenated with the predicted ab values of the colorized image. The night vision image L coefficient consists of black gradients in almost all places. This dark gradient can be combined with the green-red-blue values from the predicted image however our output image will still have a large dark gradien, making the image seem almost identical to our input image. Hence our concatenation step at the end had a flaw and hence our first model could not produce any reasonable results.

Model 2-

Input Image Colourised Image



Input Image Colourised Image Ground Truth Image

Analysis:

The results are much better than the previous two models as we were able to exploit the pretrained network. However the results do not compare very well to the ground truth. The main reason for this is the resolution of the night vision image. There is a loss of resolution which impacts the quality of colorization by the pretrained network. Methods to improve the resolution of a night vision image can be explored as a part of future work for colorization of night vision images. As this model provided reasonably satisfactory results we also performed object detection on the night vision and its corresponding colorized image to check if object detection accuracy improves. We performed object detection tests using the [Google Vision API] on a random subset of images with their ground truth, night vision and colorized versions. Results for the images did not show any significant improvement on colorization as the network can learn to detect the same features in a night vision image and a colorized image. Hence colorization does not guarantee improvement of object detection of night vision images.

Object Detection confidence	Person	Bicycle
Ground Truth	89 %	89 %
Night vision Image	92 %	94 %
colorized Image	89 %	89 %

Conclusion

In this project we proposed two methods of colorizing black-white night vision images. Our first method was based on manipulating the Lab color space of a night vision image and integrating into an encoder - decoder model using night vision-color image pairs from the KAIST multispectral dataset. However due to the large black gradient of the L coefficients from the night vision images, we obtained dark images when we concatenated the input L and predicted ab values. Then we proposed a second method based on using a pretrained black-white image colorization network. We performed image preprocessing on our night vision images and performed object detection on the resulting images to obtain acceptable colorized results. Finally, we also showed that our object detection accuracy is not significantly dependent on image colorization.

References

1. MIT DeOldify : https://github.com/jantic/DeOldify/blob/master/README.md
2. KAIST Multispectral Pedestrian Dataset : https://soonminhwang.github.io/rgbt-ped-detection/
3. colorful Image colorization (Zhang et al) : https://richzhang.github.io/colorization/
4. Floyd Hub Image colorization implementation : https://blog.floydhub.com/colorizing-b-w-photos-with-neural-networks/
5. Google Vision API : https://cloud.google.com/vision/
6. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2016). Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(1), 142-158.
7. Girshick, R.B. (2015). Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV), 1440-1448.
8. Ren, S., He, K., Girshick, R.B., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 1137-1149.
9. Redmon, J., Divvala, S.K., Girshick, R.B., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788.
10. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., & Berg, A.C. (2016). SSD: Single Shot MultiBox Detector. ECCV.

Our repository

1. Night vision Image colorization using Lab : https://colab.research.google.com/drive/1yRw4Ax79coAkQAan13p5wpwBYI6Rk3vT
2. Night vision Image colorization using inversion: https://colab.research.google.com/drive/1nZsfbEFSemcnQuuyHqZNImX7nhD2Mwvp