JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Surveillance Video Improvisation by Colouring and Enhancement of B/W video

Sarika N. Zaware[1], Abhishek R. Agarwal[2], Prachiti P. Bhagwate[3], Kaustubh H. Salunkhe[4], and Nachiket S. Suvarnakar[5]
[1] AISSMS Institute Of Information Technology, Pune
WWW home page: https://aissmsioit.org/
[2] AISSMS Institute of Information Technology Kennedy Road, Near RTO, Pune - 411 001, Maharashtra, India.

ABSTRACT:

Noise removal, resolution, contrast improvement, and colourization are all used in the colourization of cctv footage to restore the black and white film medium to its original state. In addition, most cctv footage is recorded in black and white or with low-quality colours, necessitating colourization for security reasons. We present a framework for dealing with the colourization process semi-interactively in this paper. Our research is centred on attention mechanisms and temporal cnns (convolutional neural networks). Our suggested source-reference attention allows the model to colourize long movies with an arbitrary number of reference colour images while maintaining temporal consistency without the need for segmentation. Quantitative study demonstrates that our framework outperforms present techniques and that, in assessment to present techniques, the overall performance of our framework is advanced to that of existing approaches

INDEX TERMS - TEMPORAL CONVOLUTIONS, SOURCE-REFERENCE ATTENTION, CNN (CONVOLUTIONAL NEURAL NETWORK)

1 Introduction

In today's date CCTV cameras exist all over the world and most surveillance footage is black and white as those videos take less disk space and at the same time are more efficient in capturing clear footage at night time. These footages are partly clear to view and analyze but a colourful video footage would prove more efficient for surveillance purposes, but a colour video takes more disk space in storing than a black and white video, hence a need for enhancing and colourising a black and white video footage arrives after it being recorded. There are various software and applications for the purpose of colourising video footage but most are not efficiently able to enhance, colourize and increase certain aspects of the video. In this project we actively enhance the video quality as well as colourize it while upgrading its video artifacts such as refresh rate, white-balance, noise, saturation.

1.1 Motivation

Generally, surveillance video cameras record black and white video as their primary form of data as they capture better quality during night or in low light conditions. Most of the crimes or robberies take place at night time or early mornings as there are less amount of people around. Thus, to track down the wrong doers, colourised videos would be more efficient and helpful for the authorities and so our project aims at colourizing the surveillance footage and enhancing the video quality using CNN.

1.2 Problem Definition

Unsolved cases can encourage additional people to engage in criminal activity, causing unnecessary hassle and injury to others who aren't engaged or involved in any way. This could also cause financial damage to the companies or people involved in these circumstances. With a proper colour video wherein objects, persons involved can be easily observed and noted ; authorities will be able to more effectively track down wrongdoers. The proposed system will help the authorities in tracking the responsible people.

2. Literature Survey

The main purpose of our project is to colorize and enhance the quality of surveillance videos. To achieve the same, we have referred some of the existing methods proposed by different authors. Some of them are listed below and how they helped us to reach our desired goal

[Hengyuan Zhao et al.] proposed a system where reference images are used to create color embedding which are later used for result generation using progressive feature formalization networks in his paper [1]. It works only with the help of reference images without bringing severe artifacts. Our method was inspired by the reference image method with enhancement and progressive feature formalization.

This is another reference based video colorization method which uses temporal correspondence between target frames to reflect the reference color. [Naofumi Akimoto et al.] used dense tracking method as well as instance tracking. Scene switches and camera angles changes are not handled well by [2] this method. Our method is superior to this method in terms of handling scene changes and camera angle differences.

For instance, [Satoshi Iizuka et al.] proposed a method which uses an adaptive CNN. This paper [3] presents a fully automatic colorization method using deep neural networks to minimize users' efforts and the dependency on color images. This is an adaptive clustering technique proposed to incorporate this issue. Numerous experiments state that this method outperforms state-of-the-art technology.

A deep learning and a reference color image based method is proposed by the [Mingming He et al.]. Unlike conventional deep learning models, it performs well and lets users control the result. Above paper [4] does not assist our work. We do not seek any resemblance and similarity with the given method.

This approach is based on convolutional neural networks and is able to perform the colorization without any user intervention. This method is proposed by [Hiroshi Ishikawa et al.] Color adaptation and resolution process are the key points in this paper [5]. The same model can style transfer, that is, color an image using the context of another. It uses the same architecture of image referencing and convolutional neural network.

The method proposed by [Wei-Sheng Lai et al.] relies on a deep recurrent neural network-based approach to get the enhanced video for both short and long term temporal losses. This approach [6] is agnostic to the underlying image-based algorithms applied to the video and generalizes to a wide range of unseen applications. It helps in enhancement of the video and deals with the FPS flickering.

This method [7] focuses on denoising of grayscale/color images. The models proposed by [Stamatios Lefkimmiatis] lead to very competitive results for AWGN distortions while they also appear to be very robust when the noise degrading the input deviates from the Gaussian assumption. The use of the proposed networks as sub-solvers in restoration methods that deal with more general inverse imaging problems such as deblurring, inpainting, demosaicking, etc.

This denoising architecture that extends kernel-predicting networks enables temporal and multiscale filtering. It further justifies its use in our method of denoising architecture. Source aware encoders which robustly handle diverse data from rendering systems over sampling rate. This method [8] proposed by [Thijs Vogels et al.] provides a wide range of video enhancement methods.

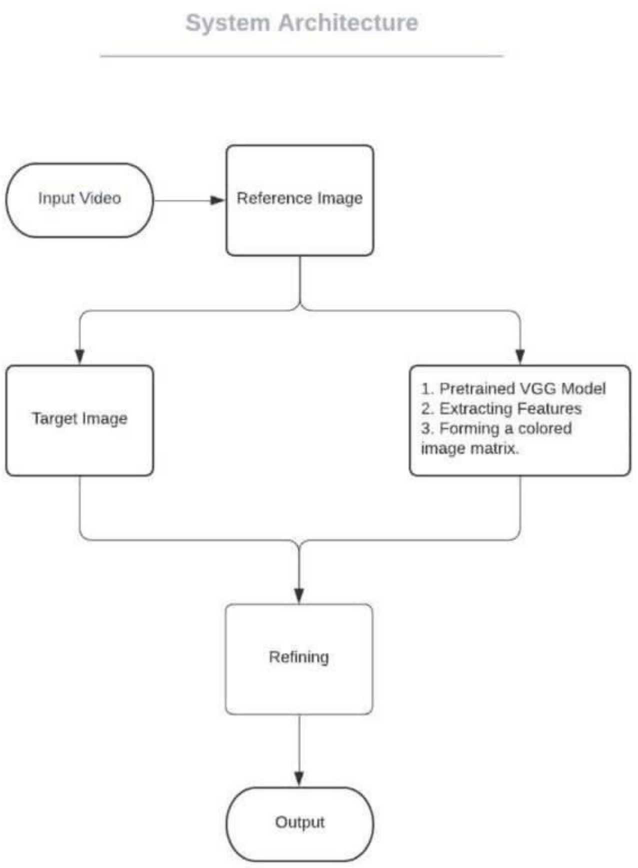
The task of video colorization is a promising signal for learning to track without requiring human supervision. [Carl Vondrick et al.] proposed a method [9] in which a colorful reference frame causes a visual tracker to automatically emerge, which we leverage for video segmentation and human pose tracking. This method was a great leap in the area of unsupervised video colorization. But our method depends on reference image colorization where it differs.

A deep CNN and a well-chosen objective function can come closer to producing results indistinguishable from real color photos. This method [10] not only provides a useful graphics output, but can also be viewed as a pretext task for representation learning. This method proposed by [Richard Zhang et al.] is mainly useful for object classification, detection, and segmentation, performing strongly compared to other self-supervised pre-training methods.

3 System Architecture

As shown in the figure below, the input can be a non-coloured or black/white video. The following phases will be applied to the given input. Video insertion, a black/white video will be uploaded. A set of Frames are extracted from the given input video. Reference Image, reference images will be uploaded as well to help in colourizing the images. Merging, the colourized images generated with reference images are then merged to form the desired output video.

Figure 1: System architecture



1)Convolutional Neural Network: They process the video frame by frame in this stage, propagating the colour from a coloured key frame to the remainder of the scene. To colourize photos automatically from black/white photographs, a learning-based model called Convolutional Neural Network (CNN) is utilised.

2) Temporal Convolution: By merging parts of RNN and CNN designs, Temporal Convolutional Networks, or TCN, is a version of Convolutional Neural Networks for sequence modelling problem

3) 3.Attention: Natural Language Translation (NLT) was the first application of attention mechanisms in neural networks. Attention for neural networks works similarly to human attention in that it allows the model to focus on different areas of the input.

4 Algorithms

4.1 Steps:

1. Data Collection: Data collection is the systematic gathering and measurement of data from a number of sources in order to obtain a complete and accurate picture of a subject area.

2. Video to Frame Conversion: For the colourization process, video is split into a certain ’n’ number of frames.

3. Image Referencing: Colourized frames created previously with help of CNN are used as Reference images for further video colourization.

4. Output Generation: Colourized video is produced for output using various video colourization models.

4.2 Algorithms And Libraries Used:

– Convolutional Neural Network: It is a type of Artificial Neural Network, and a Deep Learning algorithm which can take in an input image and assign aspects to images and differentiate.

– Temporal Convolutions: Temporal Convolutions

– is a variation of CNN for sequence modelling tasks .It can take any length sequence and map it to a sequence of the same length as an output.

– Source-Reference Attention: Source-Reference Attention is used to provide an arbitrary amount of colour reference photos that the model can utilise as colourization indications for videos.

– cv2: It is used capture the video and convert the captured video into respective frames. Eg. cap = cv2.VideoCapture(opt.input) - Load video

– PyTorch library It is a Deep Learning tensor library maily used for applications running on GPUs.

– Python PIL: It is a Python Imaging Library which provides image editing and reading capabilities. PIL.Image.open() - It is used to open and identify the given image.

– Skimage: i.e. Scikit-image, it is an open source python package used for image processing.

– Torchvision: It is used for computer vision along with PyTorch, it helps to transform and save those images.

4.3 Mathematics Associated With The Project:

We would be considering Euclidean Distance.

’h’ is height.

’w’ is width.

We will be multiplying them and raising the power to two as we have two channels ’a’ and ’b’

$$L_2\left(\hat{y}, y\right) = \frac{1}{2}\sum_{h,w}\left|Y_{h,w} - \hat{y}_{h,w}\right|^2$$

Now, the data-set used for converting the black and white images to colour images is the “image-net lightness” , the colours ”red” and ”green” are termed as ’a’ and the colours ”blue” and ”yellow” are termed as ’b’.

When user is providing a b/w image at the time, what we have is the function where we get the image in the form of h x w x 1 i.e. the height multiplied by width multiplied by 1 channel that is grey scale which is black and white.

We want the result in ’a’, ’b’ format, so we would be expecting a function in h x w x 2 , ’2’ here represents ’a’ and ’b’ , being the ”green, red” and ”blue yellow” channel. So Y cap is the function which will provide us h x w x 2 and input is h x w x 1, now to get that we are using regression with l2 function and when we use that, is seems inadequate hence not fulfilling our result

$$x = R^{h.w.1}\ \hat{y} = R^{h.w.2}$$

So, we are using multinomial classification, we are considering logarithmic scale and summing it twice

$$L(\hat{z}, z) = \frac{-1}{h.w} \sum_{h.w} \sum_{q} Z_{h,w,q} \, log(\hat{z}_{h,w,q})$$

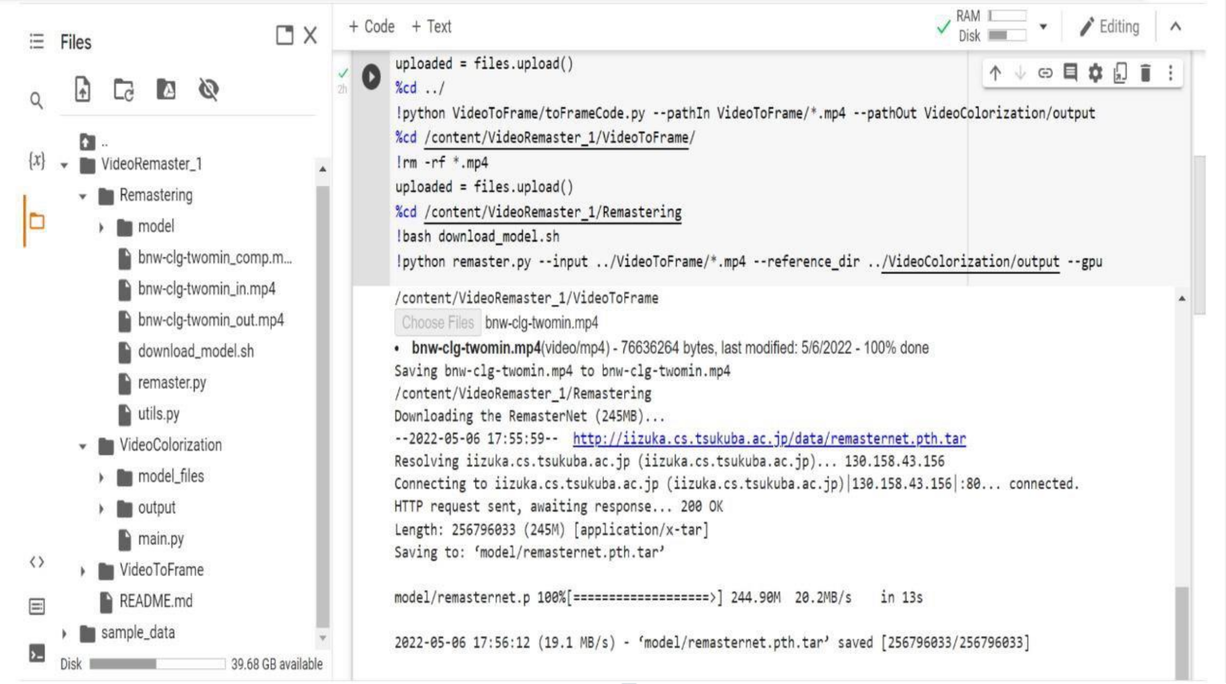
Since we take object-less images and pass it to function of all the colours which contains the bins with grid of size 10 and quantice at ’a’, ’b’. The output space keeping q = 313 as we are considering lightness as well. Thus, introducing re-balance model to achieve the optimal level of brightest in the output image. Following being the formula:

$$L(\hat{z}, z) = \frac{-1}{h.w} V(z_{h,w}). \sum_{q} z_{h,w,q} \, log(\hat{z}_{h,w,q})$$

5 Results

The output is a colourized video of previously uploaded non-colourized video, with enhanced video quality with respect to saturation, brightness and noise.

Figure 2: Colorization Process (in Google Colab environment)



```
uploaded = files.upload()
%cd ../
!python VideoToFrame/toFrameCode.py --pathIn VideoToFrame/*.mp4 --pathOut VideoColorization/output
%cd /content/VideoRemaster_1/VideoToFrame/
!rm -rf *.mp4
uploaded = files.upload()
%cd /content/VideoRemaster_1/Remastering
!bash download_model.sh
!python remaster.py --input ../VideoToFrame/*.mp4 --reference_dir ../VideoColorization/output --gpu
```

```
/content/VideoRemaster_1/VideoToFrame
Choose Files bnw-clg-twomin.mp4
• bnw-clg-twomin.mp4(video/mp4) - 76636264 bytes, last modified: 5/6/2022 - 100% done
Saving bnw-clg-twomin.mp4 to bnw-clg-twomin.mp4
/content/VideoRemaster_1/Remastering
Downloading the RemasterNet (245MB)...
--2022-05-06 17:55:59--  http://iizuka.cs.tsukuba.ac.jp/data/remasternet.pth.tar
Resolving iizuka.cs.tsukuba.ac.jp (iizuka.cs.tsukuba.ac.jp)... 130.158.43.156
Connecting to iizuka.cs.tsukuba.ac.jp (iizuka.cs.tsukuba.ac.jp)|130.158.43.156|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 256796033 (245M) [application/x-tar]
Saving to: ‘model/remasternet.pth.tar’

model/remasternet.p 100%[===================>] 244.90M  20.2MB/s    in 13s

2022-05-06 17:56:12 (19.1 MB/s) - ‘model/remasternet.pth.tar’ saved [256796033/256796033]
```

Video sets of different lengths were used and all of these were processed under default environment provided by Google Colab. Below mentioned are some points regarding the colourization of a 2-minutelong video.

– The largest video set used is of 73 mb and 2 minutes long.

– The video lengths were increased gradually by 5 seconds each time and the corresponding processing went on increasing simultaneously.

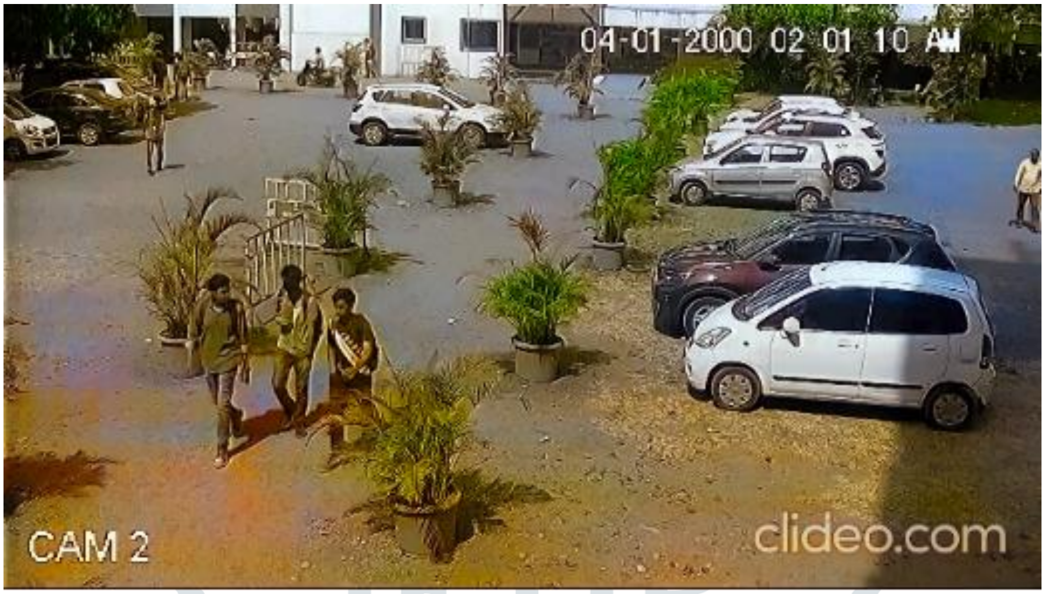
– The processing of this video took 1 hr 8 min for converting black n white to coloured video.

– It took 13 sec to generate sample images from the reference video.

– 40 sample images were generated from the reference video

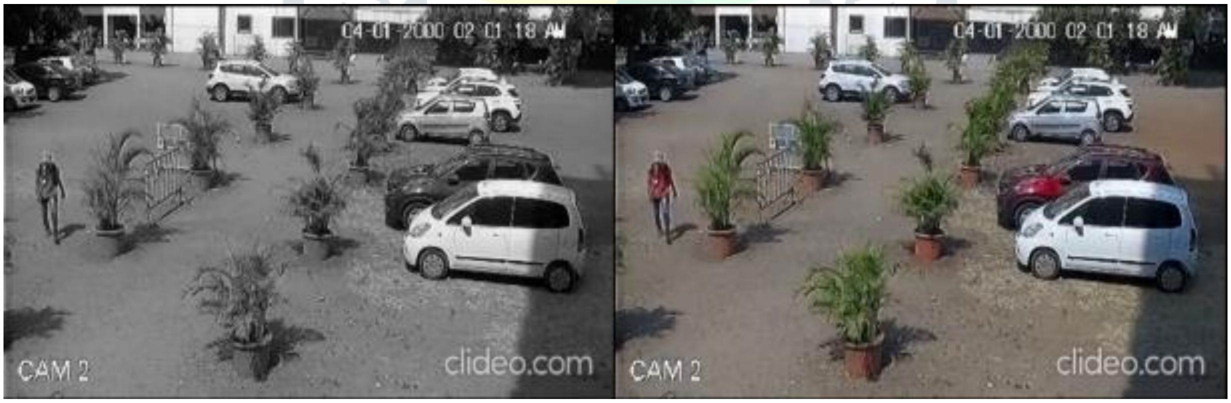
Since, the default environment here contained GPU, the video processing here could have been faster using parallel processing.

Figure 3: Frame Of Video With Absurd Values



As explained in the equation above the values of l and ab are set to 100 and 500 respectively, which generates the Fig. no. 3. In this figure, the objects with green and red colours are observed precisely while the blue and yellow colours are on lighter note since higher values of ab norm result in the same. Changing the values will determine the object colours and the training of the model will hence influence the colour of the objects in respective images. When the values of "l" (lightness) is tweaked in lower range the images will get warm colours and when the values of "l" is on the higher scale the cool colours are layered on the black and white image. The exception is raised when the abnormal values are transferred to the model via l and ab because even though it might be colouring bright images, the overall image will differ from the expected image. This will result in generation of huge delta value between expected and the actual output image obtained from the model.

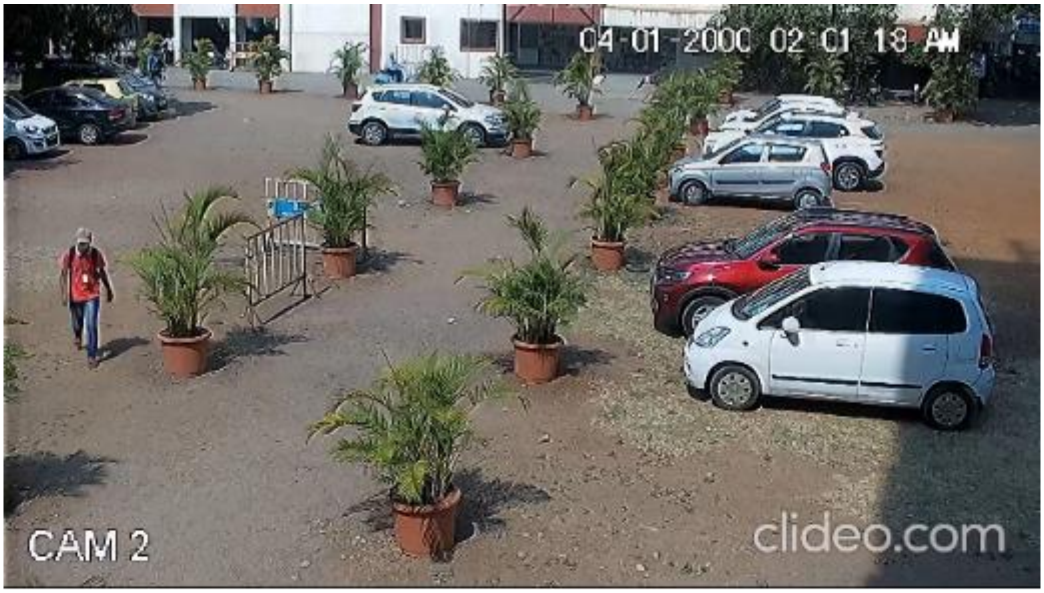
Figure 4: Differences Observed in Input and Output



As per Fig. no. 4, both black and white and it's respective colour image are being compared. The colour of the objects in the image is based on the l and ab values. The features of the reference images are taken into consideration as a matrix of tensor and is passed in the model with the black and white frames of the video given as input by the user. The model then processes the tensors and maps the values of the respective stationary object to the black and white frames.

In case of moving objects, the area of the object is calculated in terms of x and y coordinates and these are mapped onto the frames. Shadow of the object is another aspect which is taken care by l and ab factor in such a way that the model which is being trained has learnt that whenever the object is above ground level then the black shadow is present and that object is never coloured.

Figure 5: Actual Result From Model



When the expected output is compared with the actual output obtained in the video, we can observe that:

– The values of l and ab are accurate,

– The shadow constraints are also followed by the model, hence the moving object is able to obtain colours.

– We can also notice that the far background objects are not considered and are missing the features. This comes under future scope of the project i.e to obtain the features of far present objects and storing them accurately, it must also be able to transfer it to the frames of the videos.

Accuracy Of The Model:

Getting the colour surveillance video is the main scope of the project. To evaluate the model and the accuracy the values of l and ab are taken into consideration. There can be many other parameters like processing time, parallel processing, colourization of the farthest object etc but these were not covered in the scope of the project and is the area for further research. Here, the values of l and ab in actual output and expected output are compared to get the accuracy of the model. The values of expected and actual have the delta of 7.8% hence calculated accuracy of the model values comes out to be of 92.2%. Also, the entire feature transfer aspect is taken into consideration and hence the accuracy slides to 78.3%. Though this is satisfactory for the surveillance videos but can be worked upon to get the actual output close to expected output with very less computational time and higher accuracy in image objects.

6 Conclusion

The entire project is focused on colorization of the CCTV footage. The implications of this would make forensics easier and more informative and will have great in leap in security field. This project can also be useful in other fields as well for same purpose. The proposed method can be extended to a extent where it could take much less time compared to our method and also can be executed on systems with lower specifications using different algorithms without creating any video artefacts.

7 Future Work

The proposed method can be extended to a extent where it could take much less time compared to our method and also can be executed on systems with lower specifications using different algorithms without creating any video artifacts. We can also notice that the far background objects are not considered and are missing the features. This comes under future scope of the project i.e. to obtain the features of far present objects and storing them accurately, it must also be able to transfer it to the frames of the videos.

References

[1] MohammadColour2Style: Real-Time Exemplar-Based Image Colourization with Self-Reference Learning and Deep Feature Modulation, Hengyuan Zhao, Wenhao Wu, Yihao Liu, Dongliang He, Department of Computer Vision Technology (VIS), Baidu Inc. University of Chinese Academy of Sciences, 2021.

[2] Reference-Based Video Colourization with Spatiotemporal Correspondence, Naofumi Akimoto, Akio Hayakawa, Andrew Shin, Takuya Narihira, Sony Corporation, Tokyo, Japan, 2020.

[3] DeepRemaster: Temporal Source-Reference Attention Networks for Comprehensive Video Enhancement, SATOSHI IIZUKA, University of Tsukuba, Japan EDGAR SIMO-SERRA, Waseda University / JST PRESTO, Japan, 2019.

[4] Deep Exemplar-based Colourization, Mingming He, Dongdong Chen, Jing Liao, Pedro V. Sander, Lu Yuan, ACM Transactions on Graphics, Vol. 37, No. 4, Article 47. August 2018.

[5] Let there be Colour!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colourization with Simultaneous Classification, Satoshi Iizuka, Edgar Simo-Serra, Hiroshi Ishikawa, SIGGRAPH 2016.

[6] Learning Blind Video Temporal Consistency, Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, Ming-Hsuan Yang, Computer Vision – ECCV 2018 - 15th European Conference, 2018, Proceedings.

[7] Universal Denoising Networks : A Novel CNN Architecture for Image Denoising Stamatios Lefkimmiatis, In IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[8] Denoising with Kernel Prediction and Asymmetric Loss Functions, Thijs Vogels, Fabrice Rousselle, Brian McWilliams, Gerhard R¨othlin, Alex Harvill, David Adler, Mark Meyer, Jan Nov´ak, ACM Transactions on Graphics (Proceedings of SIGGRAPH 2018), vol. 37, no. 4.

[9] Tracking Emerges by Colourizing Videos, Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, Kevin Murphy, Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 391-408. 10. Colourful Image Colourization, Richard Zhang, Phillip Isola, Alexei A. Efros, ECCV 2016: Computer Vision – ECCV 2016 pp 649-666