Deep Background Generation

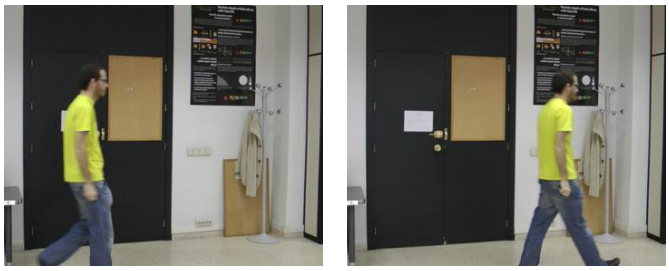
Garrett Sterling Decker, *ASU ID: 1206082081,* gsdecker@asu.edu
Jajati Keshari Routray, *ASU ID: 1211086640,* jroutray@asu.edu

Abstract—A neural network is trained to generate an image without any moving objects observed in a video. The video dataset used for training the neural network is obtained from a stationary camera with constant field of view. The project would require neural network to recognize moving objects from the array of frames, segment the background from the video and generate an image without any moving object.

Index Terms—Background generation, Background estimation, Neural Networks, Deep Learning

1 Introduction

Neural network is designed and trained to generate an image background without any moving objects observed in a video. All the pixels of the stationary background are never completely visible all the time in the video. Generating a background image from a video with constant field of view saves huge amount of data during video compression and transmission. It is also applicable in restoring old video footage shot from tape-based video camera where few frames of the video contain distortions, glitches and noise. Background generation also finds applications in video surveillance, stellar imaging and computational photography. The video dataset used for training the neural network is obtained from a stationary camera with constant field of view.



(a) (b)

Fig. 1. Sample video frames

Fig. 2. Ground truth image

2 Dataset Collection and Generation Methods:

Videos with extended scenes and have static backgrounds with moving objects are ideal candidates for data. Public domain & free to use stock footage, time-lapse videos and video dataset from ChangeDetection.net, SMBC (Scene Background Modelling Contest), SBI (Scene Background Initialization) and computer generated dataset will be used to train the neural network. Ground truth background images for training the neural network will be generated by using existing non-deep-learning methods.

3 Expected Implementation

Algorithmically, the background from a video sequence with constant field of view is generated by obtaining the median of the pixel value at every pixel from the stream of video frames. Moving objects does not change the median of pixel value at a particular location significantly. LaBGen method algorithm developed by Droogenbroeck et al. [1] generates background with very low error and uses pixel-wise temporal median filter on patches of background with moving object. The error between the generated background and background image is computed using the following benchmark parameters collected by Bouwmans et al. [2]

1) Average Gray-level Error (AGE)
2) Percentage of Error Pixels (pEPs)
3) Percentage of Clustered Error Pixels (pCEPs)
4) Peak-Signal-to-Noise-Ratio (PSNR)
5) Multi-Scale Structural Similarity Index (MS-SSIM)
6) Color image Quality Measure (CQM)

Background generation models created by Giordano et al. [3] and Chacon-Murgia et al. [4] and other papers presented in conferences use artificial neural networks to generate the background but does not use Convolutional Neural Networks. Giordano et al. [3] developed background generation method by using weightless neural networks which does not update weights of the neural networks but uses temporary RAM neurons which stores the RGB values of images random-pixel data for the sequence of video

• *Garrett S. Decker and Jajati K. Routray are with the Ira A. Fulton Schools of Engineering (CIDSE, ECEE), Arizona State University, Tempe, AZ, 85281.*
E-mail: {gsdecker, jroutray}@asu.edu

Project proposal received February 24, 2017

frames. During training, the RAM contents of the neural network are incremented (reward) by a positive number up to a maximum value and others are decremented when no change in the pixel value is observed and is passed through a function to classify the pixel and identify patterns. During background modeling, RAM neurons return RGB values at each pixel with most frequent sub pattern and generate the image.

The BE-AAPSA model developed by Chacon-Murgia et al. [4] attempts to determine a background model given a series of frames from a video by computing the average value of pixels classified as foreground in the initial 40 frames of the video. This average is used to classify the video into one of four types, each handled by a separate module. Module 1 deals with scenes with low dynamics. Module 2 deals with normal scenes which allow for moving objects. Module 3 deals with dynamic background scenes where the threshold values used to separate foreground from background are unclear (uses Fuzzy C-Means algorithm to estimate threshold values). Module 4 solves very dynamic scenes (such as jittering) by using SURF to realign frames. Reclassification into a different module type can occur part way through if a drastic change occurs in the scene (to do this, foreground is closely monitored). Once a module has been selected, pixels in each frame are divided into three classes, represented in a matrix A(x, t), where x is location and t is time or frame. This matrix A(x, t) has an effect on the learning rate of the background estimation. The background is estimated by iteratively calculating a weight for each pixel for each frame, WBM(x, t). Future weights per pixel depend on the previous iteration, as well as the learning rate, whose formula depends on which module was chosen, and which class the pixel currently belongs to. Generally, the first few frames have a large effect on the weights, as the learning rate follows an exponential decay formula.

We attempt to create a background generation model in our project using Convolutional Neural Network and Recurrent Generative Adversial Network to generate the background image. Generative Adversarial Networks are created by having two networks compete against each other. One network is a discriminator, and one network is a generator. Both networks are trained simultaneously. The discriminator is usually convolutional, and the generator is usually deconvolutional. The discriminator network takes input from both a real dataset and generated data (from the generator network), one sample at a time, and must determine the probability of the sample coming from the real dataset or not. The generator network takes noise as input, and generates data similar to the real dataset. However, the generator does not look at the real dataset for training. Rather, the generators loss function is the complement of the discriminators loss. This means that the harder it is for the discriminator to distinguish between the real and generated data, the lower the generators loss will be. Eventually, the competition between both networks will make the generator adept at generating data that looks like real data.

Suppose we have a dataset of timelapse videos, and their corresponding backgrounds. Instead of having the generator take exclusively noise as input, we can add our time lapse videos as input. The generators job would be to generate an image of the background. The discriminator would take a timelapse video as input along with a corresponding background image. The discriminators job would be to determine whether the background image came from a real dataset or our generator. If both networks get really good at doing their jobs, then we will have a generator that can take a video as input and generate a corresponding background image.

4 SCHEDULE

Project schedule	Month	Project member
Dataset generation and collection	February	Jajati
Image dataset preprocessing	March	Garrett
Deep literature analysis	March	Garrett & Jajati
Neural network architecture setup	March	Jajati
Experimentation & parameter tuning	April	Garrett
Result analysis	April	Garrett & Jajati
Project report preparation	April	Garrett & Jajati

REFERENCES

[1] B. Laugraud, S. Piérard, and M. Van Droogenbroeck, "Labgen: A method based on motion detection for generating the background of a scene," *Pattern Recognition Letters*, 2016.

[2] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1168–1177, 2008.

[3] M. De Gregorio and M. Giordano, "Background modeling by weightless neural networks," in *International Conference on Image Analysis and Processing*. Springer, 2015, pp. 493–501.

[4] G. Ramirez-Alonso, J. A. Ramirez-Quintana, and M. I. Chacon-Murguia, "Temporal weighted learning model for background estimation with an automatic re-initialization stage and adaptive parameters update," *Pattern Recognition Letters*, 2017.

[5] T. Bouwmans, L. Maddalena, and A. Petrosino, "Scene background initialization: a taxonomy," *Pattern Recognition Letters*, 2017. [Online]. Available: http://sbmi2015.na.icar.cnr.it/SBIdataset.html

[6] M. Braham and M. Van Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," in *Systems, Signals and Image Processing (IWSSIP), 2016 International Conference on*. IEEE, 2016, pp. 1–4. [Online]. Available: https://orbi.ulg.ac.be/bitstream/2268/195180/1/Braham2016Deep.pdf

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] D. J. Im, C. D. Kim, H. Jiang, and R. Memisevic, "Generating images with recurrent adversarial networks," *CoRR*, vol. abs/1602.05110, 2016. [Online]. Available: http://arxiv.org/abs/1602.05110