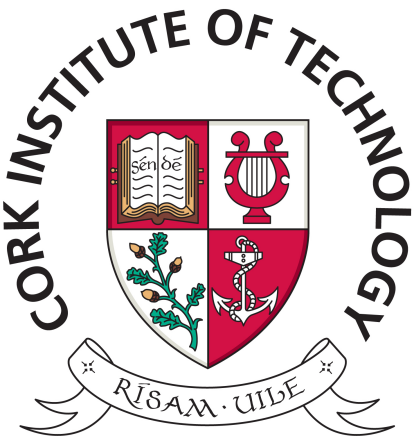


Anomaly Classification using Ensemble Learning over CCTV camera footages

by
Shrey Mishra

This proposal has been submitted in partial fulfillment for the module Research Practice and Ethics

in the
Faculty of Engineering and Science
Department of Computer Science

January 5, 2020

Declaration of Authorship

I, Shrey Mishra, declare that this thesis titled, Anomaly Classification using Ensemble Learning over CCTV camera footages and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for an masters degree at Cork Institute of Technology.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Cork Institute of Technology or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this project report is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.
- I understand that my project documentation may be stored in the library at CIT, and may be referenced by others in the future.

Signed: _____

Date: _____

Abstract

In this project, we will discuss about the use of CCTV cameras for detection of Crimes and Accidents based on feeding the raw real time CCTV footage to an Ensemble voting classifier. The classifier consists of a series of machine learning algorithms and Deep learning architecture with special preference to the proposed State of the Art Capsule Networks by Geoffrey Hinton in 2017.

The motive of the project is to drastically reduce the crime and accidents caused using live surveillance and provide timely assistance. Apart from the moral aspect, we will also study the algorithmic improvements that can be added to the current state of the art solutions.We will further improve it upon adding our own significant solution, on which we will compare our result and the current state of the art solution in the domain.

The visualization aspect of the data is extremely crucial before evaluation, alongside measuring the performance of each of the individual algorithm / architecture to produce a combined 'Ensemble Classifier' and finally measuring the run-time performance and the loss of the proposed classifier.

Acknowledgements

Contents

Chapter 1

Introduction

CCTV crime detection is a very crucial application of video surveillance system, where it has been evidently seen that by simply using the CCTV cameras , there was a reduction in crime rate in specific regions, each of them having a different impact depending upon the demography.

To explain the importance of CCTV camera alone for monitoring, We will be referring to the subtle conclusions made on a recently released study titled **"CCTV surveillance for crime prevention"** published in **March 2019** conducted by the **Cambridge University** upon the funding from **the Swedish National Council for Crime Prevention.** The review is very important and different from any other source for reference as it is very precise in highlighting the importance of CCTV after a 40 year research and meta analysis of the crimes using evaluation metric as Odds Ratio (OR). This study is a benchmark to convince that simple addition of CCTV camera without AI monitoring can drastically reduce the crime and moreover adding the AI element can be a game changer which will be discussed in the second study.

Study 1(Cambridge University): It was found that a 16 percent reduction in crime was associated with CCTV, which was a significant effect. With a greater impact on car parking areas decreasing the crimes by a massive 51%. In the meta analysis section of the paper, the authors have released an odds ratio (OR) which directly correlates with the crime reduction. An OR greater than 1 indicates that the use of CCTV camera had a significant, desirable impact in reducing the crime

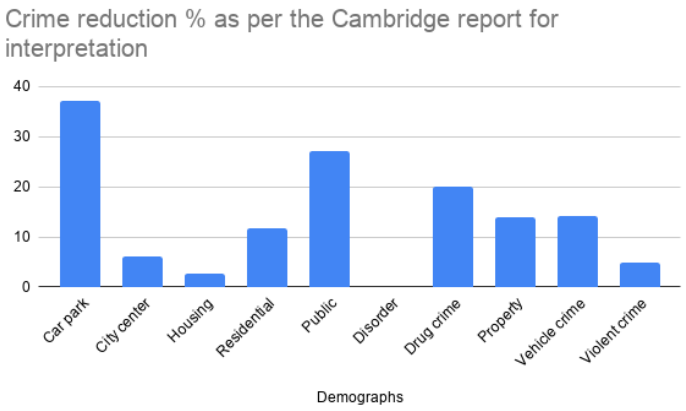


Figure 1.1: conversion of odds ratio to crime reduction rate

in that particular demography.
The graph provided in the Figure 1.1 is a direct interpretation of the results produced in the study[8].

Study 2: According to the another study[1] conducted in **2002**, the evidences indicate and correlate that In Burnley, the crime began to reduce approximately one month before the cameras had been installed indicating that the intent of simply being watched could create a fear of getting caught in the minds of the criminals. This is where, I believe the proposal could come handy when the AI can be used as an assistance tool on top of the video monitoring for timely actions, ultimately resolving the crimes and accidents in real time.The real time aspect of the solution could cause fear among the criminals and ultimately reducing the crime percentage. Unfortunately there is no study currently done on the use of AI based video surveillance due to lack of the hardware and the processing power. These assumptions are subtle and based on the indications from the previous studies done in the domain.
In my opinion this presents an excellent evidence that using AI powered CCTV monitoring algorithms can effectively and timely control the situation.

In this proposal we will be trying to implement anomaly classification task while training on the raw images feed from the CCTV camera. In the practice we will be using the **UCF crime dataset** [**13**] available that has 128 hours of video surveillance and classified into 13 realistic anomaly classes. The data set

contains very large number of instances which is ideally good for any deep learning algorithm to work on. We begin the analysis by visualizing the images of a 2D/3D plot by applying dimensionality reduction technique for visualization called **TSNE** as introduced by the Geoffrey Hinton [**?**]. The sole purpose of the visualization is to visually show that the separation of the anomaly present image is very different from the one in which the anomalies are absent. The sheer presence of cluster of anomaly present images will be very different from the cluster of normal frames.Further application of **Kohnen's Self Organizing Map [3]** will help us visually plot the topology of the map clearly showing that most of the anomaly points lie far from the converged map. The Combination of both the visual techniques will actually help us in understanding of the data, meaning that the anomaly classes are very well separated from the normal category of images. Both of the above approaches are set to yield good results when set for a large number of iterations as both of these algorithms are unsupervised by nature. It is important to understand that these algorithms will clearly play no role in the detection accuracy of the final model and there sole purpose is to provide a strong proof of concept only. Before we actually begin exploring the data set and applying various classification algorithms.
The final result is based upon the Ensemble classifiers consisting of various CNN architectures(ResNet,GoogleNet,VGG18 etc) in conjunction to the Capsule Networks algorithms.

Note- We could also include some of the other Machine Learning algorithms depending upon the choice and as per the supervisor in charge, but in this proposal I will mainly focus on just two.

The entire procedure can simply be broken into smaller sections namely:

- Extraction of anomaly frames and labelling them, as part of preprocessing
- Visualizing the data on a SOM map trained over TSNE
- The split of Data into training and testing or a K-fold approach.
- Applying individual algorithms and evaluating the loss
- Making of the Ensemble classifier
- Evaluating the Ensemble classifier based on run time and loss measure

Chapter 2

Background

Most of the background analysis of this section will be based on **Waqas Sultani's paper [12]** titled "Real-world Anomaly Detection in Surveillance Videos." The paper was recently published in 2019 and informs about the background of the previous State of the Art.The authors have proposed a novel ranking loss function known as MIL. To the best of my knowledge there are currently no other papers citing the dataset as the dataset was originally created by the Sultani's team. The comparison of any other paper in domain would create a bias in the measure as changing the dataset would drastically change the performance metric results, so instead in this proposal ,I will explain Sulatni's work and further extend it with my possible assumptions that can significantly improve the performance.

Study 1(Validation of dataset:): He has extensively evaluated his dataset with many other datasets in the category and his seems to have the maximum number of frames and length.

	# of videos	Average frames	Dataset length	Example anomalies
UCSD Ped1 [27]	70	201	5 min	Bikers, small carts, walking across walkways
UCSD Ped2 [27]	28	163	5 min	Bikers, small carts, walking across walkways
Subway Entrance [3]	1	121,749	1.5 hours	Wrong direction, No payment
Subwa Exit [3]	1	64,901	1.5 hours	Wrong direction, No payment
Avenue [28]	37	839	30 min	Run, throw, new object
UMN [2]	5	1290	5 min	Run
BOSS [1]	12	4052	27 min	Harass, Disease, Panic
Ours	**1900**	**7247**	**128 hours**	**Abuse, arrest, arson, assault, accident, burglary, fighting, robbery**

Table 1. A comparison of anomaly datasets. Our dataset contains larger number of longer surveillance videos with more realistic anomalies.

Figure 2.1: why the dataset is better than the other proposed datasets in the domain

Method	AUC
Binary classifier	50.0
Hasan *et al.* [18]	50.6
Lu *et al.* [28]	65.51
Proposed w/o constraints	74.44
Proposed w constraints	**75.41**

Figure 2.2: Auc score comparison's on all the previous work done in the domain

The figure 2.1 is directly taken from Sultani's paper where he compares his dataset with many standard datasets used for evaluating models such as the UCSD crime which are very popular in the domain.The data quality is evidently better and larger than the other proposed standard datasets.

Chronological Order - The chronological order of the improvements made in the State of the art approach are as follows:

Lu et al [5] suggested a dictionary-based approach for learning normal behaviours and used errors in reconstruction to classify anomalies.PCA (Principal component analysis) was applied using the sparse matrix representation.**Hasan et al [2]** proposed a fully convolutional feed- forward deep auto-encoder based approach to learn local features and classifier.

Sultani proposed a 3D CNN architecture with a 60 % dropout (tested empirically) which is the current state of the art and will later be discussed in this section when compared to my solution.

The Figure 2.2 is an image extracted from Sultani's paper actually highlights the AUC score (area under the curve) for the previous state of the art approaches.As a baseline model, he is using the Binary classifier (An SVM based linear kernel approach) in his paper.

Challenges discussed in Sultani's paper- In Sultani's paper, the authors have clearly highlighted 2 of the failure cases in the section **(g) and (h)** of the paper. Which will again be a setback for most of the deep learning algorithms including ours.

- The absence of light is a possibility where the algorithm fails to flag anomalies (e.g.- A man entering the dark house from the window) and sometimes raising false alarms when insect crawls on the video feed.

- Large Normal public gathering was subsequently misclassified into a false alarm.

Scope of Improvement-In addition to the current proposed limitations there are quite a few algorithmic challenges in his adopted approach that possibly yield an even better score if optimised.These challenges are aimed to be solved with this proposal. We will be aiming to point out the scope of improvement in his approach taking his approach as our base model and further highlighting the limitations in his approach.

- Use of many popular CNN architectures such as RESnet and GoogleNet instead of just one in the detection accuracy as an Ensemble model which has the potential to greatly benefit instead just using one of the detection architecture/algorithm. The idea can be fundamental but empirically proven in fabio's paper[7].Another real world example is the Imagenet challenge **ILSVRC 2015** where the the top 1-12 models were ensemble as to a single proposed architecture which ranked 13th on the challenge.

- The conceptual flaw of CNN failing to understand the spatial geometry could be a problem. Since the cameras are usually placed at awkward angles (on top in the outdoors), correlating with anomalies would require a spatial understanding of features, this could be a breakthrough in his approach as Geoffrey Hinton [11], theoretically proved the concept on one of his dataset called CIFAR 10 dataset which actually consists of many images of cars.

- In the paper Sultani investigates sigmoid and relu activation function but has no mention of tanh or a popular leaky Relu which has a significant impact over the training time and have no zero slope problem, which can potentially improve the scores. Another paper titled **"SEARCHING FOR ACTIVATION FUNCTIONS" [9]** shows the significant advantage of using other optimization functions such as Swish and Leaky Relu, The results are tested on the Image-net dataset. These are simple one line changes in the code of activation function but can significantly improve the performance by roughly 1%.

- The optimiser Adagrad used in the Sultani's paper does not take into account the history and hence can take longer time to converge. In practice there are two commonly known optimiser that usually perform a faster and a better convergence in low number of iterations , namely RMSprop and Adam optimiser. Theoretically explained Adam optimizer alone has the advantage of both RMSprop and Adagrad. A recent paper published in April 2019 [10], empirically proves the advantage of Adam and further compares adam with the current state of the art AMSgrad improving and converging faster than the current Adam used in many of the underlying deep learning architectures

- There are many approaches where the initialisation of the random weights actually play a very important role in deciding how many iterations it will take the algorithm to converge to the global minima. Sultani's paper does not speak anything about initialisation of weights which I assume to be random. In practice the approach followed is usually the "HE" initialization for Relu and Xavier initialization for Sigmoid or Tan hyperbolic based activations [4]. In the context there are many other initialisation

- There could be an added visualization to differentiate that anomaly frames extensively differ from the normal video frames.(Please see the methodology section).

- Authors have used just one of the measure to evaluate their models, over 4 cross fold validation, In practice we can use many other evaluation metrics to get a much truer insight for bench marking the performances. e.g.- F1 score and log loss measure

Sultani's deep learning architecture is performing really well with a 75% AUC score, However his approach can be further improved by using the adopted measures that clearly indicate a significant performance boost in both the run time (training time) as well as the prediction accuracy. There are several design flaws in his architecture and can be algorithmically solved by adding more complex CNN architectures.

Chapter 3

Proposal

For the implementation we will use the python programming language version 3.6 as many of the library dependencies do not work with python 2.7. The main libraries used for the implementation of the algorithms will be:

- Scikit learn for implementation of machine learning algorithms
- Keras uses tensor flow backend for deep learning models
- Pickle to extract and save the model
- Matplotlib and seaborn for exploratory data analysis
- Pandas for preprocessing the data

The dataset contains 13 anomaly classes for each of those videos we will categorise them into two bags where the positive bag contains the image frames of anomalies and then the negative bag contains the normal frames. The classifier will then recognise if the new given image corresponds to anomaly or not.

On top of the classifier there will be a second classifier that will detect which type of anomaly it corresponds to. In Sultani's paper both the bags images are feed to the 3D convolutional neural network that identifies the presence of anomaly by a metric called anomaly score, they have changed the problem context to a regression problem where each of the frame is given a certain value, known as the anomaly score. Based on the score we analyse the given frames and conclude if the given output was anomaly or not. However, I will treat this problem as a classification problem, where there are two classifiers running simultaneously where the first classifier will have it's weight trained to detect abnormality of

the frames extracted while the second classifier will categorise if the results were positive (anomalous) from the first classifier. To train the model we will NOT be using all 13 classes and only a smaller subsection of classes would be used (depending upon the supervisor in charge), However In the scope of improvements discussed in the background section mostly relate with the dataset Cifar 10 and hence many categories involving a group of people might not be a good a criteria for selection of anomalies as pointed in Sultani's paper where his model fails to detect when there are too many people in the frame and raises a false alarm. Recommended choice of subcategories entirely depends on the empirical accuracies obtained after training the network but a good recommendation of subcategories would be

- Arson
- Road accidents
- Fighting
- vandalism

To begin our analysis we will employ the data visualization techniques to show that the anomaly classes are very distant from the normal classes. Since the data usually consists of image frames having many pixels in each frame(roughly 1 million), to visualise the data we will employ TSNE algorithm available in Scikit learn that can be used to effectively understand higher dimensional data to denote classes. An example of the demo visualization[6] obtained on training TSNE with the MNIST dataset (consisting of hand written digit images) is given below:

We can see there is a formation of clusters (with very little overlap), Each digit is denoted with a unique colour in the 'tsne' plot and was successfully able to converge with an ideal perplexity value(also known as a hyper parameter). The visualization aspect only shows us the cluster formation but fails to represent the general topology of the data.

To better understand the topology of the data an additional Self organising map will be deployed to show that most of the data frames lie in a particular region except the ones that are anomaly.The below image shows the self organising map trained on the PCA extracted features.

The main idea is to understand and visually show the separation of anomaly points from the regular ones, Additionally we could separate each of the frames

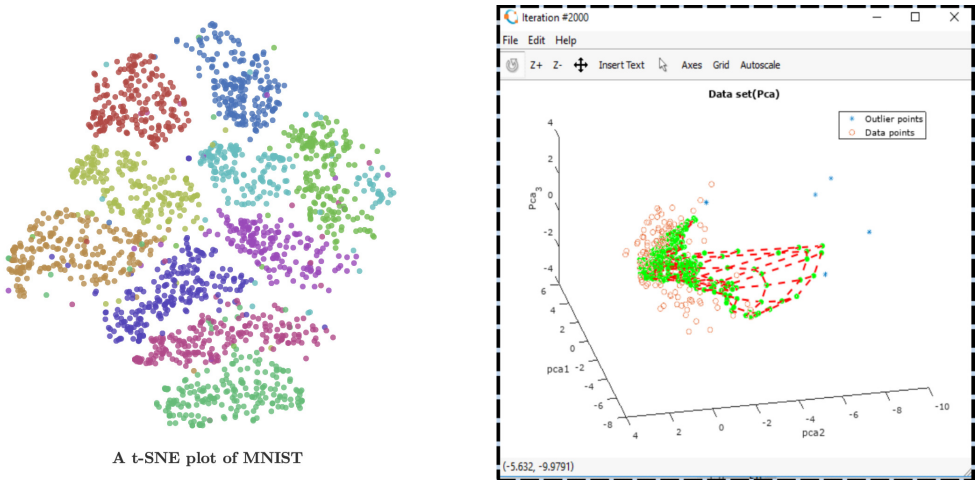


Figure 3.1: TSNE used for visualizing MNIST and Self organizing map trained over PCA shows the topology of the data points**(image data)**

depending upon the anomaly in another TSNE plot that physically differentiates the types of anomalies. Dimensionality reduction in the deep learning domain is very subjective as using many popular approaches such as PCA disrupts the features and for that very reason we will use auto encoder model, where the number of hidden layer and input nodes would be found empirically to be best suited for the data. Usually for image classification task there are mainly two algorithms. (more like SVM's can be added depending upon the approval from the supervisor)

- Convolutional Neural Networks (multiple architectures)
- Capsule Networks

For CNN's there exist many different type of architectures available, usually the more hidden layers we have, the better is the detection up until there is gradient vanishing problem in the network if the model is too deep.

The final model would be an Ensemble classifier (voting classifier), that takes many different CNN architectures along side the Capsule Networks to make a combined prediction.Each of the individual weak Lerner in the ensemble model is given a weight associated with the performance of the model on training, The more is the associated weight the greater is the algorithm's stake in predicting the Ensemble classifier.

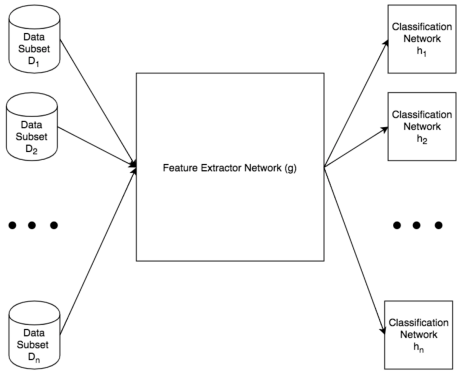


Figure 3.2: Ensembles of various deep learning algorithms/architecture

The challenge here is to construct different types of individual weak learners with little or no overlap (rarely possible) so that there is enough variation of neural nets given to capture most of the aspects of the data.

Figure 3.3 explains and gives a rough idea of how ensembles work with the Deep learning classifiers.

The CNN architecture has fully connected deep neural network hence, the weights assigned are the hyper parameters here. They are trained using the Back-propagation where the weights are updated by an optimiser, in our case it will be Adam optimiser with Gradient Descent as it usually performs better and faster convergence over the proposed Sultani's Adagrad approach.

The weights will be initialised using Xavier or He initialisation depending upon the activation function used. For Tanh or sigmoid we prefer Xavier and for Relu or leaky Relu we prefer the He initialisation.

The weight update will take place in batches because we cannot update weight after every sample as this is computationally expensive for the classifier to train, hence the choice depends upon the training time available for the scope of the project. Usually Stocastic gradient descent is used when we want to update weights after every sample point in the data. The rough mini batch size of 32 or 16 (when using mini Batch gradient descent) performs almost equally in terms of accuracy(empirically proven) with a much faster training time.

A list of CNN architectures and algorithms to be used with the Ensemble classifiers are

- VGG 18/16

- Microsoft's ResNet50
- Dense Net
- Google Net

Alongside the state of the art Capsule Networks,Both TSNE and Self organizing maps are unsupervised algorithms that do not require any complex hyperparameter tuning, however the capsule and CNN's will require us to run backpropogation for many number of iterations also known as epochs in the deep learning domain or until convergence.

3.1 Ethical considerations

Have a public license associated with the dataset with freedom to add and modify the data as per need, along side author's approval to use his dataset received on personal email and can be reproduced whenever needed.

Chapter 4

Methodology

Evaluation metrics and Comparison- The Evaluation of an individual model will be based upon the mean score obtained from the AUC, F1 score and the log loss over a 10 fold cross validation. The 10 fold validation assumes that each of the sample is passed in the training and testing phase.

Sampling of the dataset - Each of the sample will be processed such that for a given subcategory of anomalies have a class balance for example, the cases for vandalism should be roughly equal to the cases of explosion. This will avoid the class imbalance and any bias towards one class by the network.

Visualisation of the data- As mentioned in the proposal we will use the TSNE and Kohnen's self organising maps.

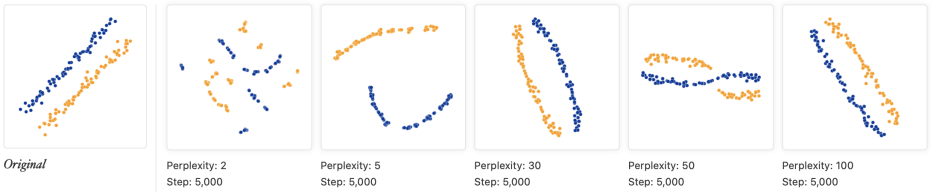
How TSNE works ?

- Step 1- calculating the similarity between each of the data points by entering each of the data point over a gaussian distribution and recanting other points on the curve. By renormalising for all points we calculate the probability of each point being similar to the chosen data point over which we plotted gaussian distribution.
- Step 2- This time we try plot a Cauchy distribution over each of the data-point and again recalculate the probabilities as done in step 1.
- Step 3- We use the KL divergence to map probability sets obtained from

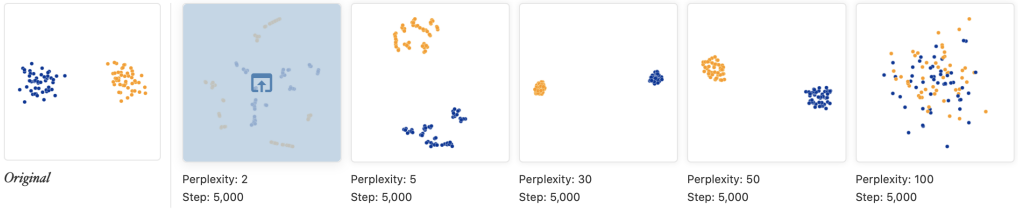
both the steps to map a high dimensional data and finally we minimise the cost function of KL divergence using the gradient descent algorithm.

Why we choose TSNE over PCA-

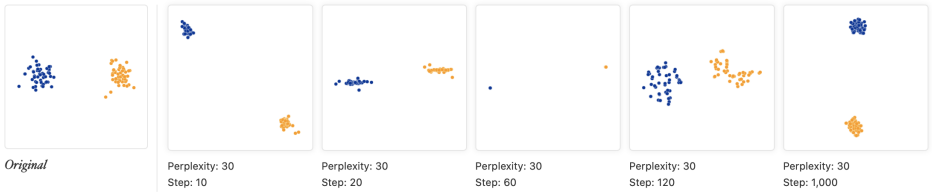
- PCA is a fairly old algorithm that tries to preserve the data over the axis that maximise variance where as TSNE preserves the neighbourhood.
- PCA works well with linear data which is mostly not the case with images as they have complex symmetry, while TSNE works well with non linear data (for e.g. cylindrical, spherical etc)



TSNE tries to preserve the neighbourhood between the sample and roughly associates them into clusters(however varies after every run), these clusters can be interpreted as a bunch of data points with similar features.The algorithm takes perplexity value as a hyper parameter. Which is empirically found.



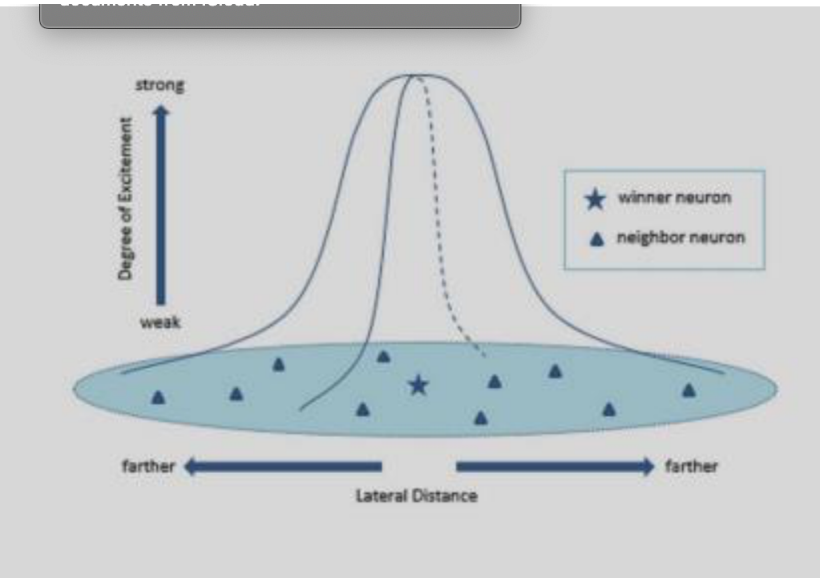
After selecting the right perplexity value we will then focus on the iterations required for the gradient descent algorithm to converge.



Self organising Maps-

- **Initialisation:**All the neurons on the SOM map are initialized at the beginning with small value random weight vectors.

- **Sampling:**At each iteration a point from the input sample is chosen and weights of the SOM map are tweaked depending upon this input
- **Competetion:**The chosen input sample will try to find it's best matching neuron on the SOM map by calculating the euclidean distance.
- **Cooperation:**The winner neuron will then influence each of the nearby neurons in the vicinity by pulling them towards it.
- **Adaptation:**based on the influenced neighbourhood neuron we will adapt or update the weights associated for each of the iteration run , and hopefully in the set number of epochs we will converge or will repeat until the map is converged.



Evaluation is based upon the Ensemble model of various CNN architectures and Capsule Networks.

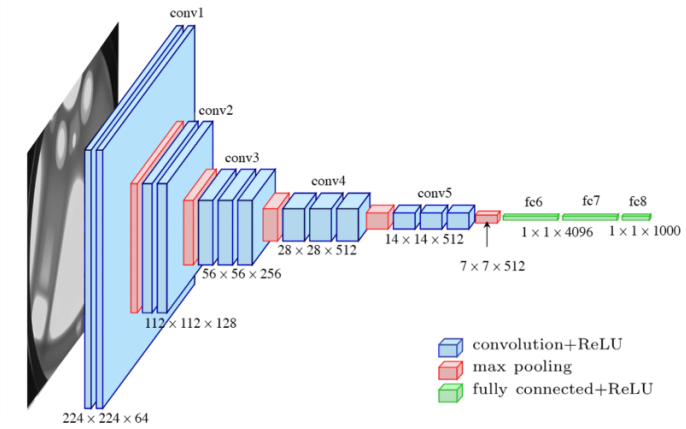
The CNN architectures follow a common methodology with each of them having the 3 basic stages with different assignment(orientation)

- Convolution filter to extract information based upon the padding and stride provided
- Pooling layer, usually a max pooling layer is applied on top of the convolution filter to get the most dominant features for learning
- Fully connected deep neural network in the End- The main motto of CNN that differentiates it from ANN is the lower number of features obtained by

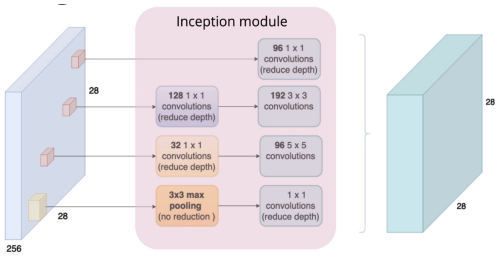
the different assignment of many convolution and pooling layers that speed up the training.

The architecture of the proposed CNN's are as follows

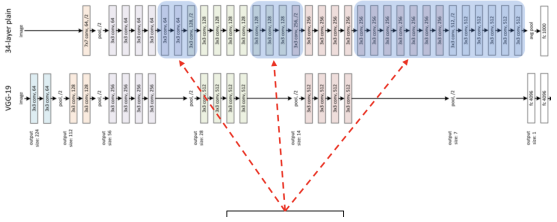
VGG 18-



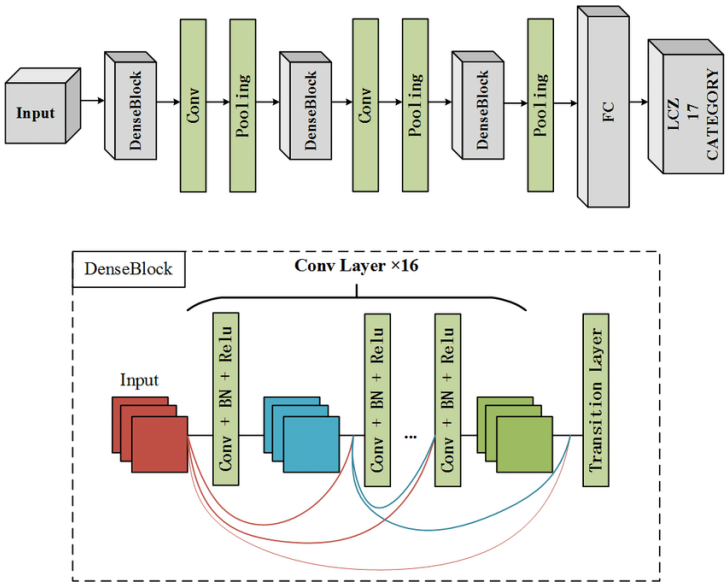
Google Net-



Resnet -



Dense Net-



Capsule networks-

CNNs generally work by accumulating sets of features at each layer. It starts off by finding edges, then shapes, then actual objects. However, the relationship between the spatio-temporal features is lost while using CNNs. For example, while identifying a face, a mere presence of two eyes, one nose and one mouth is enough for a CNN to classify it as a face, the presence of these objects in the wrong place does not affect the accuracy of the CNNs. As per Geoffrey Hinton et. al. "The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster." The introduction of the capsule network gives us capability to take the advantage of spatio-temporal relationship between features.

In convolutional capsule layers the output of each capsule is nothing but the local grid of vectors to each type of capsules in the above layer which can be calculated using different matrix transformation for each member of the grid and for each type of capsule.

Chapter 5

Work plan

The proposal of the work explained is mainly divided into three stages for execution and 2 additional for evaluation and write-up

Stage 1 (Data preprocessing)- Structuring the video footages into frames of anomalies and not anomalies, Further classifying each of the anomaly class to the subcategory of anomalies for the second classifier to work upon. The process would generally require time directly proportional to the data since the dataset has 128 hours of video recordings I assume categorising them would take at least a month's time. The data is already preprocessed into categories of videos, however the frame extraction has to be still applied to categorise the data into bags.

Stage 2 (Visualisation of the dataset)- In this section we are aiming to prepare the data to be ready for visualization and showing that a decision boundary (visually) exists for separation of the two classes.

For this task we already proposed that we would be using TSNE and Self Organising maps.

With TSNE the implementation can be achieved within a week (due to Scikit learn's internal implementation).

For Self organising map we don't really have a working implementation of the algorithm in the scikit learn hence we have to make our own implementation or have to use any third party dependencies.

In my previous internship (France) I implemented the concept from scratch in the Octave programming language, We can reuse the same octave code by

exporting the data produced in a **.mat** extension. To our benefit libraries such as scipy can help us achieve the task fairly easily.

However if we wish to write the entire code in python or look for dependencies that help us achieve the implementation it could take roughly unto two weeks(could take more).

Some of the dependencies found that could be useful are:

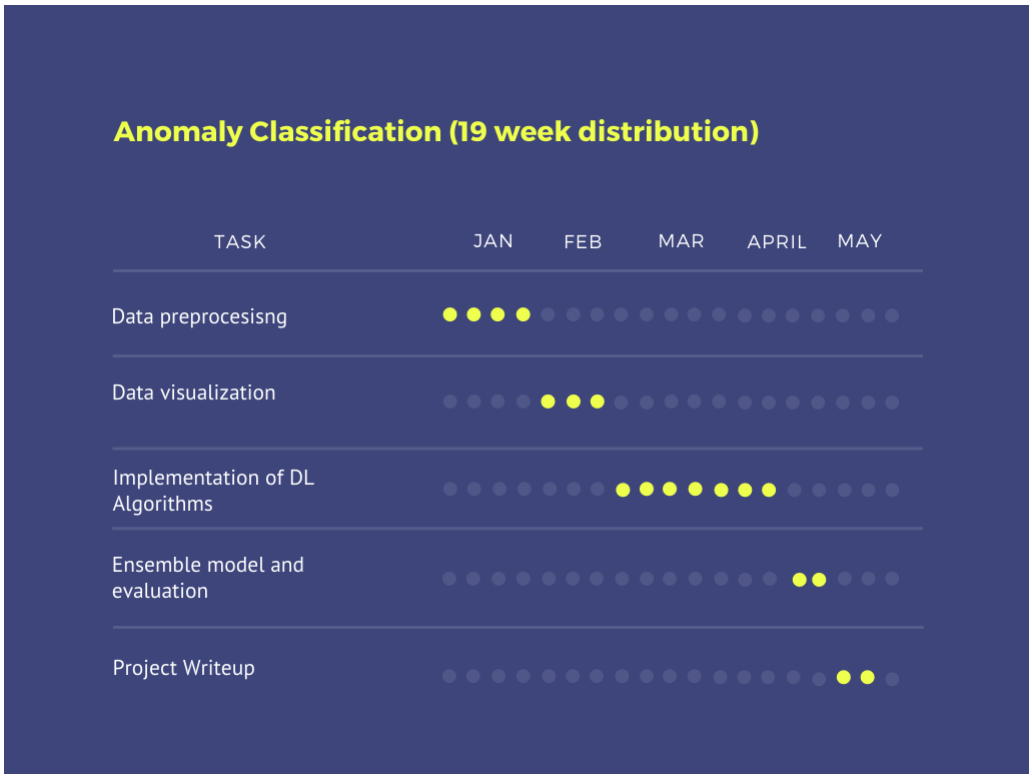
Stage 3 (Implementation of Deep learning)- This stage comprises of execution of the various Deep learning models and together putting them into an Ensemble classifier. Classical CNN architectures proposed in the plan consist of mainly four different types, implementing them very vastly depends upon the familiarity with the Keras library (Yet to be discussed in Semester 2 with deep learning module), However I assume that overall this should take not more than a month.

Secondly the use of the Capsule Networks with the algorithm heavily depends upon the the understanding of the original paper Dynamic routing between capsules and currently has very limited support to the online available code. We are most likely going to construct our own code from a python class instance. This can take unto 3-4 weeks.

Stage 4 (Validation and making up of the final Ensemble)-The last part is creating the Ensemble of the model and saving all the previous models into a new model and evaluating the powerful ensemble model on the data. I assume this should take about 2 weeks because we would already have the individual weak learners ready.

Stage 5 (Thesis Write up)-Lastly and most importantly the write up of the project report and a detailed explanation of each of the models (upto 20k words I assume) would take a 2 week time till it's ready.

Below is work-plan Gantt chart:



The project has a span of 18 weeks starting the development from January itself, the project should be ready by the mid of May and maximum extending up to the end of May.

Chapter 6

Summary

The given proposal illustrates a working state of the art solution to the problem of Anomaly classification for CCTV recordings, with an end to end solution. The deep learning models could be reused with other image classification tasks. The proposed model are leveraging out a rigorous visualisation, evaluation and micro analysis (hyper parameters and optimizers) of deep learning model.The expected outcomes discussed in the improvement section should surely improve the performance with a lesser training time.

At the end I would like to say "No amount of money can compensate for an individual's loss of life." If this model is able to detect these common anomalies and save just one life with a timely action, then there is no other motivation stronger than this in my opinion.

Bibliography

[1] Rachel Armitage. To cctv or not to cctv. *A review of current research into the effectiveness of CCTV systems in reducing crime*, 8, 2002.

[2] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742, 2016.

[3] Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[4] Siddharth Krishna Kumar. On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863*, 2017.

[5] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *Proceedings of the IEEE international conference on computer vision*, pages 2720–2727, 2013.

[6] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[7] Fábio Perez, Sandra Avila, and Eduardo Valle. Solo or ensemble? choosing a cnn architecture for melanoma classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[8] Eric L Piza, Brandon C Welsh, David P Farrington, and Amanda L Thomas. Cctv surveillance for crime prevention: A 40-year systematic review with meta-analysis. *Criminology & Public Policy*, 18(1):135–159, 2019.

[9] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[10] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

[11] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *Advances in neural information processing systems*, pages 3856–3866, 2017.

[12] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2018.

[13] WaqasSultani. Ucf crime dataset, 2019.