Team Gabru

ANOMALY DETECTION IN VIDEO FEEDS

Machine Learning Term Project

Team Name: Team Gabru

Project Supervisor: Prof. Pabitra Mitra

Project Mentor: Anirban Santara

Aishik Chakraborty | 13CS30041
Ashish Sharma | 13CS30043
Chinmaya Pancholi | 13CS30010
Jatin Arora | 13CS10057
Jeenu Grover | 13CS30042
Prabhat Agarwal | 13CS10060

ANOMALY DETECTION IN VIDEO FEEDS

INTRODUCTION

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These non-conforming patterns are often referred to as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities or contaminants in different application domains. Some of the challenges for the task of anomaly detection are:

1. Drawing the boundary between normal and anomalous behavior
2. Availability of labeled data
3. Noisy data

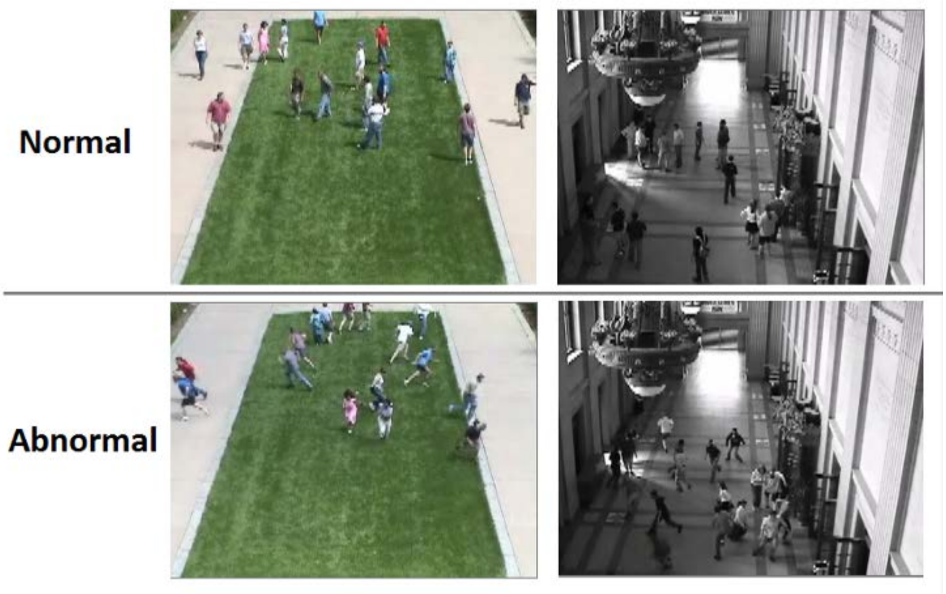


Figure 1 Anomaly

- ### TYPES OF ANOMALY

Anomalies can be classified into following three categories

1. **Point Anomalies** - An individual data instance can be considered as anomalous with respect to the rest of data.
2. **Contextual Anomalies** - A data instance is anomalous in a specific context (but not otherwise), then it is termed as a contextual anomaly (also referred as conditional anomaly). Each data instance is defined using following two sets of attributes :

- **Contextual attributes** - The contextual attributes are used to determine the context (or neighbourhood) for that instance. Eg: In time- series data, time is a contextual attribute which determines the position of an instance on the entire sequence
- **Behavioral attributes** - The behavioral attributes define the non-contextual characteristics of an instance. Eg: In a spatial data set describing the average rainfall of the entire world, the amount of rainfall at any location is a behavioral attribute

3. **Collective Anomalies** - A collection of related data instances is anomalous with respect to the entire data set.

• OUTPUT OF ANOMALY DETECTION

Anomaly detection have two types of output techniques:

1. **Scores:** Scoring techniques assign an anomaly score to each instance in the test data depending on the degree to which that instance is considered an anomaly.
2. **Labels:** Techniques in this category assign a label (normal or anomalous) to each test instance.

• APPLICATIONS OF ANOMALY DETECTION

1. **Intrusion detection:** Intrusion detection refers to detection of malicious activity. The key challenge for anomaly detection in this domain is the huge volume of data. Thus, semi-supervised and unsupervised anomaly detection techniques are preferred in this domain.
2. **Fraud Detection:** Fraud detection refers to detection of criminal activities occurring in commercial organizations such as banks, credit card companies, insurance agencies, cell phone companies, stock market, etc. The organizations are interested in immediate detection of such frauds to prevent economic losses.
3. **Medical and Public Health Anomaly Detection:** Anomaly detection in the medical and public health domains typically work with patient records. The data can have anomalies due to several reasons such as abnormal patient condition or instrumentation errors or recording errors. Thus the anomaly detection is a very critical problem in this domain and requires high degree of accuracy.
4. **Industrial Damage Detection:** Such damages need to be detected early to prevent further escalation and losses.
5. **Image Processing:** Anomaly detection techniques dealing with images are either interested in any changes in an image over time (motion detection) or in regions which appear abnormal on the static image. This domain includes satellite imagery.
6. **Anomaly Detection in Text Data:** Anomaly detection techniques in this domain primarily detect novel topics or events or news stories in a collection of documents or news articles. The anomalies are caused due to a new interesting event or an anomalous topic.
7. **Sensor Networks:** Since the sensor data collected from various wireless sensors has several unique characteristics.

Objective

In this project, we aim to **identify anomalies in video feeds** using Machine Learning techniques.

Related Work

There are three fundamental approaches to the problem of outlier detection:

- **Type 1:** Determine the outliers with no prior knowledge of the data. This is essentially a learning approach analogous to unsupervised clustering. The approach processes the data as a static distribution, pinpoints the most remote points, and flags them as potential outliers. Type 1 assumes that errors or faults are separated from the 'normal' data and will thus appear as outliers. The approach is predominantly retrospective and is analogous to a batch-processing system. It requires that all data be available before processing and that the data is static. However, once the system possesses a sufficiently large database with good coverage, then it can compare new items with the existing data. There are two sub-techniques commonly employed, diagnosis and accommodation (Rousseeuw and Leroy, 1996) [1]. An outlier diagnostic approach highlights the potential outlying points. Once detected, the system may remove these outlier points from future processing of the data distribution. Many diagnostic approaches iteratively prune the outliers and fit their system model to the remaining data until no more outliers are detected. An alternative methodology is accommodation that incorporates the outliers into the distribution model generated and employs a robust classification method. These robust approaches can withstand outliers in the data and generally induce a boundary of normality around the majority of the data which thus represents normal behaviour. In contrast, non-robust classifier methods produce representations which are skewed when outliers are left in. Non-robust methods are best suited when there are only a few outliers in the data set as they are computationally cheaper than the robust methods but a robust method must be used if there are a large number of outliers to prevent this distortion. Torr & Murray (Torr and Murray, 1993) [2] use a cheap Least Squares algorithm if there are only a few outliers but switch to a more expensive but robust algorithm for higher frequencies of outliers.
- **Type 2:** Model both normality and abnormality. This approach is analogous to supervised classification and requires pre-labelled data, tagged as normal or abnormal. Classifiers are best suited to static data as the classification needs to be rebuilt from first principles if the data distribution shifts unless the system uses an incremental classifier such as an evolutionary neural network. A type 2 approach can be used for online classification, where the classifier learns the classification model and then classifies new exemplars as and when required against the learned model. If the new exemplar lies in a region of normality it is classified as normal, otherwise it is flagged as an outlier. Classification algorithms require a good spread of both normal and abnormal data, i.e., the data should cover the entire distribution to allow generalisation by the classifier. New exemplars may then be classified correctly as classification is limited to a 'known' distribution and a new exemplar derived from a previously unseen region of the distribution may not be classified correctly unless the generalisation capabilities of the underlying classification algorithm are good.

- **Type 3:** Model only normality or in a very few cases model abnormality (Fawcett and Provost, 1999) [3], (Japkowicz et al., 1995) [4]. Authors generally name this technique novelty detection or novelty recognition. It is analogous to a semi-supervised recognition or detection task and can be considered semi-supervised as the normal class is taught but the algorithm learns to recognise abnormality. The approach needs pre-classified data but only learns data marked normal. It is suitable for static or dynamic data as it only learns one class which provides the model of normality. It can learn the model incrementally as new data arrives, tuning the model to improve the fit as each new exemplar becomes available. It aims to define a boundary of normality. A type 3 system recognises a new exemplar as normal if it lies within the boundary and recognises the new exemplar as novel otherwise.

GENERAL APPROACH

We can view abnormal behavior detection as a type of high level operation of image understanding, where logical information is extracted from input image sequences and used to model behavior. The figure shows a sketch of the general process.

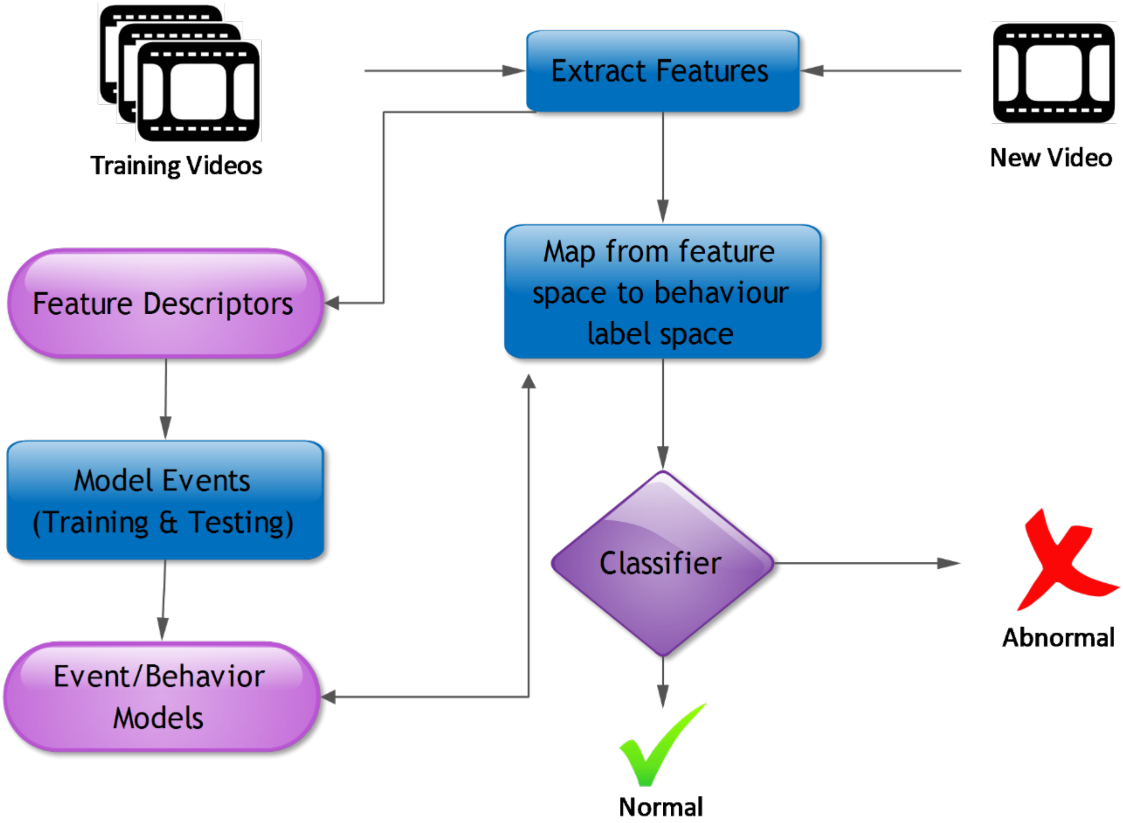


Figure 2 General Approach

A. Behavior Abstraction and Representation

The abstractions used to represent events and construct behavior model is the first step of an anomaly detection system. The abstractions are either pixel based (pixel-level description with primitives such as gradient, color, texture, motion history image, etc.) or object based (object-level description with primitives such trajectory, size, shape, and speed of object). Other commonly used object based abstractions are bounding boxes, blobs, and silhouettes.

In the same vein, features that are used to encode behavior can be “global” or “local,” either spatial or temporal or both. Spatial–temporal features have shown particular promise in motion understanding due to its rich descriptive power and, therefore, widely used as a feature descriptor.

Information such as trajectory, speed, and moving direction are often captured from moving objects. Optical flow is quite commonly used. When motion-information and spatial-location features are used to describe behavior, spatial deviations and unusual speed or inactivity are considered abnormal.

B. Modelling & Learning Framework

1. **Supervised:** These methods build models of normal or/and abnormal behavior based on the labeled data. Video segments that do not fit the models are “flagged off” as abnormal. This modeling approach for unusual events’ detection is good only if these abnormal events are well defined and there are enough training data. The challenge although in this is how to incorporate long term scene adaptation.

2. **Unsupervised:** These methods utilize the concepts of co-occurrence statistics on extracted features from unlabeled video data. They learn the normal and abnormal patterns from the statistical properties of the observed data. Isolated clusters identified as anomalies. Normal, Poisson, and other distributions are used for statistical modeling for normal patterns. Variants of HMMs, the Bayesian modeling framework, topic models, etc. can been employed to make probabilistic inference on unseen data. Hence these methods keeps updating their model of the world and “normal” behaviour as more and more data is seen and hence are adaptive to scene changes.

3. **Semisupervised:** These approaches fall in-between the first two. They learn a model of usual/unusual events using partially labeled data at either the features level or the clips level. Bayesian adaptation techniques can also be used to create models for unusual events in an unsupervised manner.

DATASET

We used UCSD Anomaly Detection Dataset [5] for our experiments. It was acquired with a stationary camera mounted at an elevation, overlooking pedestrian walkways. The crowd density in the walkways was variable, ranging from sparse to very crowded. In the normal setting, the video contains only pedestrians. Abnormal events are due to either:

- the circulation of non pedestrian entities in the walkways

- anomalous pedestrian motion patterns

- **Dataset Composition**

  - **Dataset is divided into 2 subsets** - Ped1 and Ped2
  - **Ped1:** Contains **34 training** and **36 testing** videos
  - **Ped2:** Contains **16 training** and **12 testing** videos
  - 10 clips from Ped1 and 12 from Ped2 are provided with manually generated pixel level binary masks.

APPROACH 1: Using Optical Flow

This first approach we used for anomaly detection was extracting spatial-temporal features using optical flow and training classifiers for detecting anomaly. This approach is based on the paper by Saligrama et al [6]. We have used local statistical aggregates in this approach for determining anomalies in the videos. Anomalies in many video surveillance applications have local spatio-temporal signatures, namely, they occur over a small time window or a small spatial region. The distinguishing feature of these scenarios is that outside this spatio-temporal anomalous region, activities appear normal.

To analyze the video, we divide it into atoms of size 10x10x5, where 5 is the temporal dimension and the 10x10 is the spatial dimension. We then extract various features from our data.

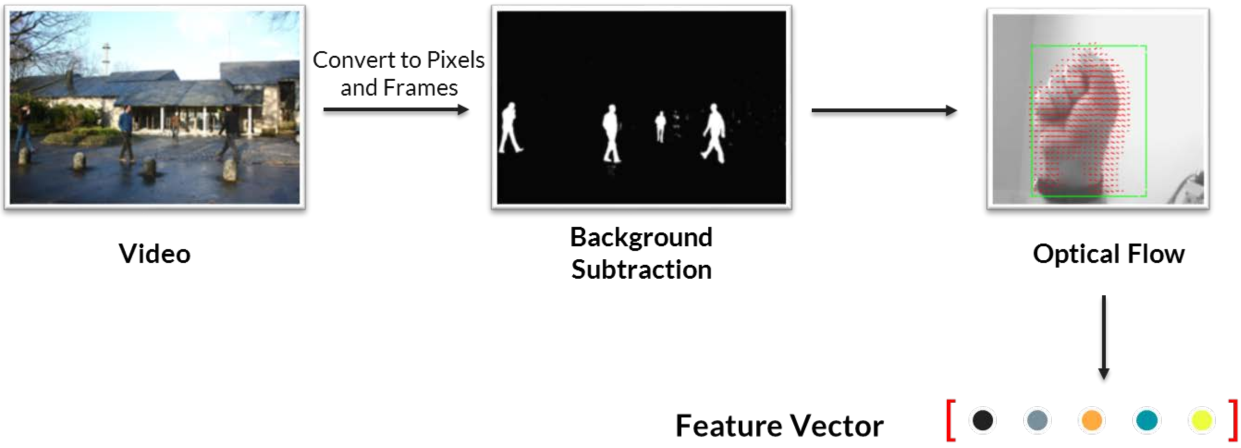


Figure 3 Approach 1: Overview

Feature Descriptors

We now describe local features that are associated with each node (atom) of our graph. During feature extraction we compute a feature value for each pixel. Then, the pixel-level features are condensed into a multidimensional vector for each atom by averaging each feature component over all the pixels within the atom.

We use the following local features:

(1) **Persistence**: Activity is detected using a basic background subtraction method. The initial background is estimated using median of several hundred frames. Then, the background is updated using the running average method. We flag each pixel as part of the background or foreground. Persistence, for an atom, is the percentage of foreground pixels in the atom.
(2**) Direction**: Motion vectors are extracted using Horn and Schunck's optical flow method. Motion is quantized into 8 directions and an extra "idle" bin is used for flow vectors with low magnitude. The feature for each atom is a 9-bin un-normalized motion histogram. The value for each bin corresponds to the number of pixels moving in the direction associated with the bin.

(3) **Motion Magnitude**: Magnitude of motion vectors for each bin (except the idle bin) is computed and averaged over all the pixels in the atom.

We thus have an 11-dimensional descriptor for each atom. While our setup is sufficiently general and admits other descriptors we use only these in our project.

Classification

After getting the feature vector for all the atoms in our dataset, we want to classify them into normal or anomalous. To do so, we trained various models on our dataset, such as:

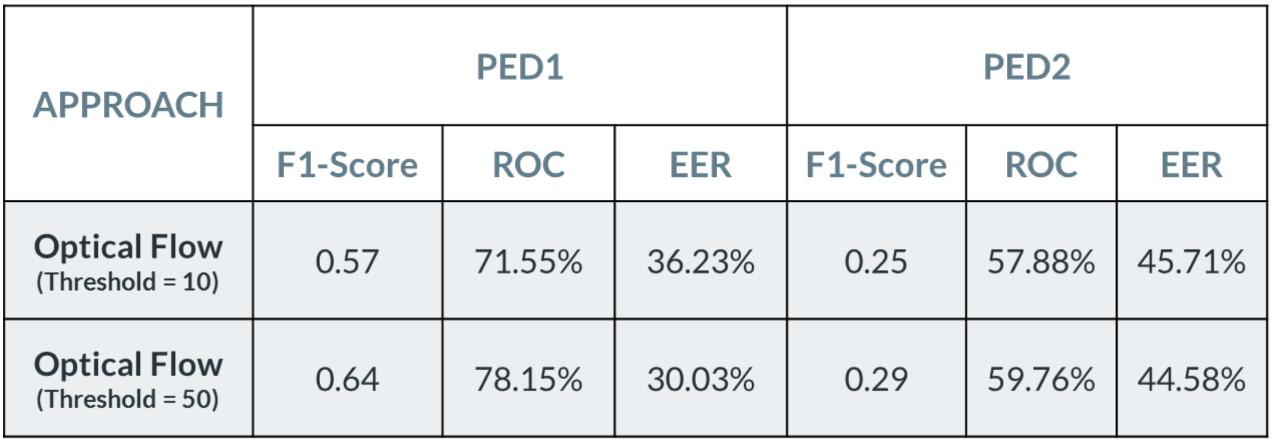
- Naive Bayes
- Random Forests
- SVM (with different kernels)
- Decision Tree
- K-Nearest Neighbours

But out of all these, **Decision Tree model** was found to give the best performance.

Experiment and Results

We used USCD Ped1 and Ped2 for our experiment. The data that we had was labelled at pixel level but we are ruuning our classifier at the atom level. To label the atoms, we made use of the constituent pixels. We kept a threshold *Th*, any atom containing anomalous pixels greater than *Th* was labeled as anomalous. We performed our experiment for the 2 values of *Th*, 10 and 50. The results are given below:

Table 1 Optical Flow Results



APPROACH	PED1			PED2		
	F1-Score	ROC	EER	F1-Score	ROC	EER
Optical Flow (Threshold = 10)	0.57	71.55%	36.23%	0.25	57.88%	45.71%
Optical Flow (Threshold = 50)	0.64	78.15%	30.03%	0.29	59.76%	44.58%

APPROACH 2: Using AlexNet

Alexnet contains eight learned layers, five convolutional and three fully-connected layers. The output of the last fully-connected layer is fed to a 1000-way softmax which produces a distribution over the 1000 class labels which are for the 1000 classes in the ImageNet Dataset the Alexnet is trained on.

The kernels of the second, fourth, and fifth convolutional layers are connected only to those kernel maps in the previous layer which reside on the same GPU. The kernels of the third convolutional layer are connected to all kernel maps in the second layer. The neurons in the fully-connected layers are connected to all neurons in the previous layer. Response-normalization layers follow the first and second convolutional layers. Max-pooling layers follow both response-normalization layers as well as the fifth convolutional layer. The ReLU nonlinearity is applied to the output of every convolutional and fully-connected layer.

Reducing Overfitting in Alexnet

The network has 60 million parameters and as such with the size of the dataset at hand, it is difficult to learn all the parameters properly without doing some amount of overfitting.
Overfitting is prevented by:

1. **Data Augmentation:** Done by artificially enlarging the dataset using label-preserving transformation
2. **Dropout:** Dropping out neurons in the first two fully-connected layers with probability 0.5.

Extracting Features

We first take a replication of the model described in the AlexNet publication with some differences:

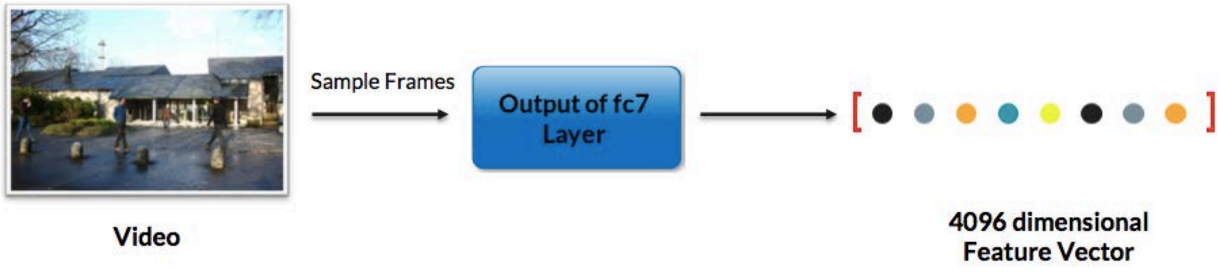


Figure 4 Approach 2: Using AlexNet

- Not training with the relighting data-augmentation;
- The order of pooling and normalization layers is switched (in CaffeNet, pooling is done before normalization).

This pretrained model is available in Caffe Models [7].

This model is trained on a 1.2M image ILSVRC-2012 dataset.

This model is snapshot of iteration 310,000. The best validation performance during training was iteration 313,000 with validation accuracy 57.412% and loss 1.82328. This model obtains a top-1 accuracy 57.4% and a top-5 accuracy 80.4% on the validation set, using just the center crop.

We use the model on the UCSD Dataset. Each video in the dataset is divided into frames. We load the pretrained Alexnet and apply it to each frame of the video. Then we take the output of the fc7 layer which gives us a 4096 dimensional vector. Such feature vectors taken as output from the fc7 layer are considered to be good representations of the frames and are used in several tasks like Image Captioning and other related tasks. This motivated us to take the output of fc7 layer as representation for each frame.

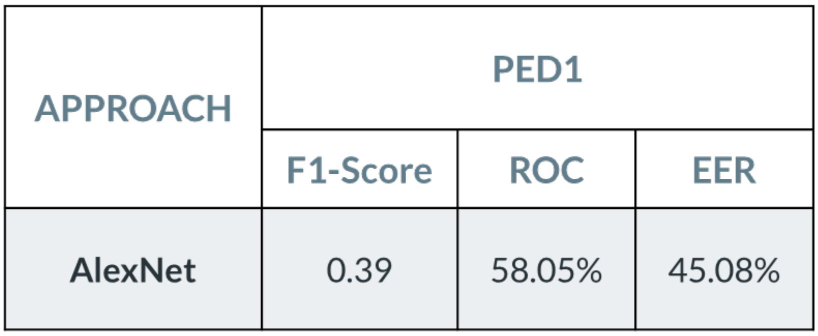
Learning a model

Now that we have feature representations of each frame, we use classical Machine Learning classifiers like Support Vector Machines, Random Forest, Naive Bayes, etc. to learn a model and evaluate it on a training dataset.

Evaluation

The Naive Bayes Classifier gave the best F1-Score and thus we are reporting it and comparing our result with our previous approach.

Table 2 AlexNet Results



APPROACH	PED1		
	F1-Score	ROC	EER
AlexNet	0.39	58.05%	45.08%

Although using traditional Machine Learning Classifiers does not perform well on the features extracted using Alexnet (F1-Score=0.39) as compared to the performance of the classifiers on the features extracted using Optical Flow with threshold as 50(F1-Score=0.64), the features extracted gives rise to a Time Series data of video frames on which other statistical methods can be applied to detect outliers/anomalies as will be discussed in Approach 3.

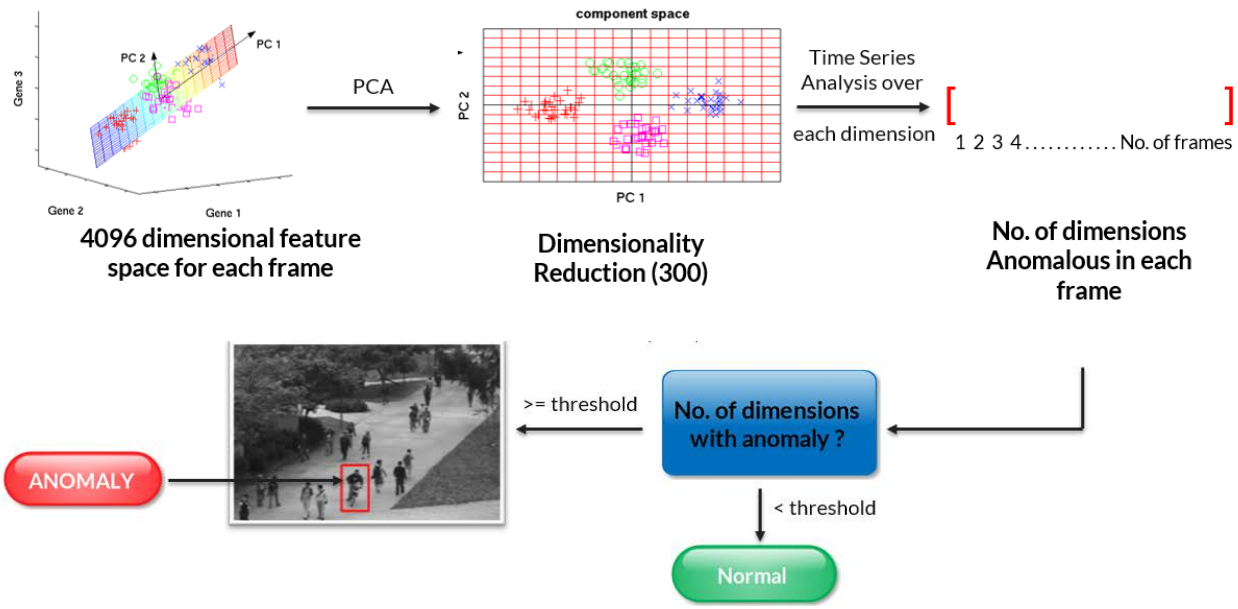
APPROACH 3: Using Time Series Analysis

We saw that by extracting the features from the FC7 layer of AlexNet, we get a good and informative representation of a video frame. So, earlier we modelled our problem as a binary classification problem and trained standard classifier using scikit-learn library in Python, like Naive Bayes, Decision Trees, K-Nearest Neighbours but we unfortunately could not get good precision and recall. Hence, here we explore standard statistical techniques for anomaly detection in the feature space we obtain from AlexNet.

Seasonal Hybrid Extreme Studentized Deviates (SH-ESD)

Generalized Extreme Studentized Deviates (ESD) is a well-established statistical procedure for detection of outliers in real values time-series data, where the assumption is that inliers are normally distributes. SH-ESD is an extension of the ESD approach which breaks down data series into piecewise approximations and can account for seasonality in the real-time data.

Experimental Setup



Gene 3
PC 2
PC 1
Gene 2
Gene 1
4096 dimensional feature space for each frame
PCA
component space
PC 2
PC 1
Dimensionality Reduction (300)
Time Series Analysis over each dimension
[1 2 3 4............No. of frames]
No. of dimensions Anomalous in each frame
No. of dimensions with anomaly ?
>= threshold
< threshold
ANOMALY
Normal

Figure 5 Anomaly Detection using Time Series Analysis

From the FC7 layer of AlexNet we get 4096 dimensional vectors for each video frame. But since we wish to apply SH-ESD for anomaly detection here, which works on only real valued data, so we consider each dimension as a separate real-valued time-series. So, we first reduce the 4096 dimensional feature space into 300 dimensional space using Principal Component Analysis (PCA). Then, we apply SH-ESD on each dimension of the reduced feature space. For SH-ESD implementation, we used Pyculiarity, a python library which is a wrapper over Twitter Anomaly

Detection Package in R which uses this algorithm. Hence, we get to know anomalous data frames as per each dimension. Now, we accumulate the results for each dimension to get number of anomalies detected in each frame. If the number of anomalies in each data frame exceeds a threshold, then we classify this frame as anomalous.

Using the above approach on 20 test videos from labelled UCSD Dataset, we obtain:

Precision: 0.67

Recall: 0.60

F1-Score: 0.63

APPROACH 4: Utilizing Topic Models

Generative topic models like Latent Dirichlet allocation (LDA), probabilistic latent semantic analysis (pLSA), etc. have shown good performance in modelling complex scenarios with a simple data representation. They have been successfully applied to automatic text analysis in the information retrieval and language-modelling domain to discover the main themes or topics from large corpus of text documents. In text analysis, a topic refers to a set of consistently co-occurring words in the text documents.

However, these topic models can also be applied to videos. In video analysis, each topic correspond to the behaviors that are frequently occurring in the scene, where the meaning of a behavior depends on the **visual words** which have been used to build the documents.

Topic Models for Anomaly Detection

The basic idea behind using topic models for anomaly detection is as follows. We first learn a generative model of typical behavior using good discriminative features, and then detect and classify abnormal behaviors (outliers) as those that are badly explained by the learned model. In this generative model framework, we know the examples of rare behaviors and build explicit models of rare versus typical behaviors.

Visual Words

Given a set of frames of a video we first find a feature vector for each of those frame by either using Optical flow or AlexNet. We cluster together these feature vector using K-means. Each of these cluster represents an activity of the videos. We extract out a representative of each of these clusters. The representative of a cluster is defined as the centroid of that cluster. Each of these cluster centers (representative) then form a visual word and we define our vocabulary as the collection of all such visual words. For a new frame, we find the cluster to which it belong and use the centroid of that cluster as the corresponding visual word. In this way, we represent each video (which is a collection of frames) as a bag-of-words.

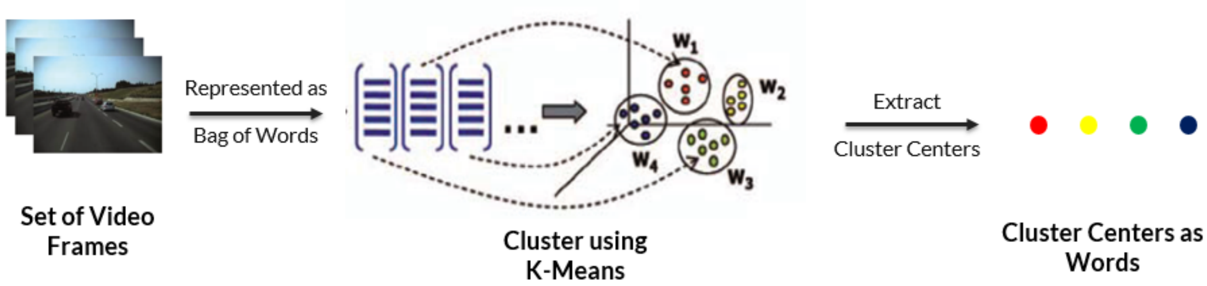


Figure 6 Topic Models: Visual Words

Learning Topic model

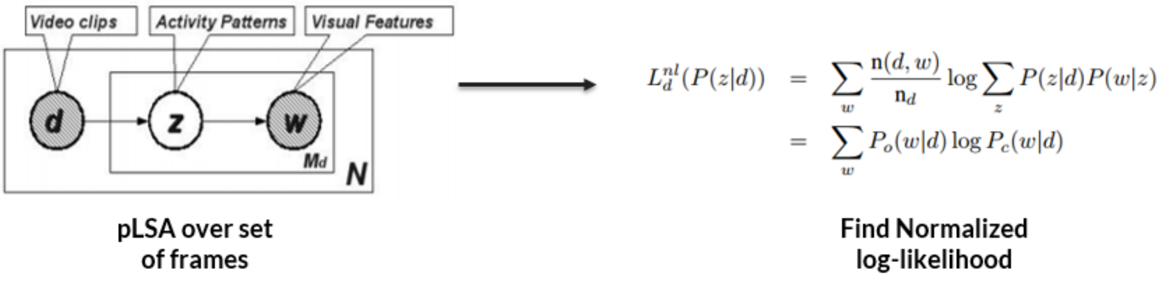
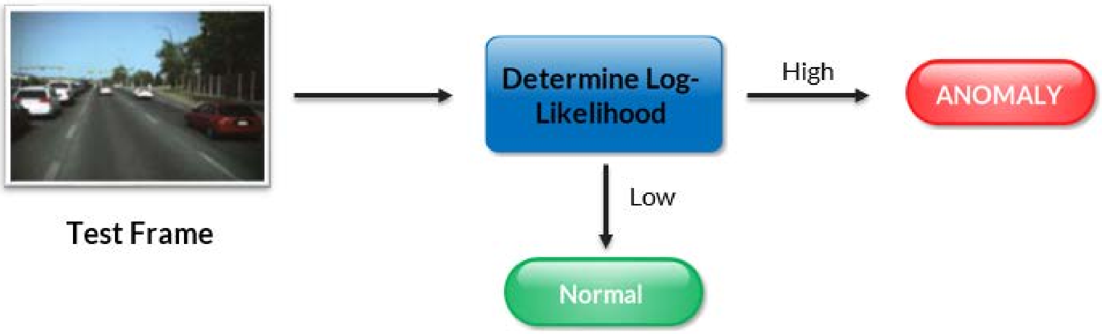

Video clips
Activity Patterns
Visual Features
d
z
w
Md
N
pLSA over set of frames
$L_d^{nl}(P(z|d)) = \sum_w \frac{n(d,w)}{n_d} \log \sum_z P(z|d)P(w|z) = \sum_w P_o(w|d) \log P_c(w|d)$
Find Normalized log-likelihood

Figure 7 Topic Models: Learning

We use probabilistic latent semantic analysis as for learning topics. pLSA [8] model originates from a statistical view of LSA. It is a statistical model that associates a latent variable z ∈ Z = {z1, . . . , zNA } with each observation (occurrence of a word in a document). These variables, usually called topics, are then used to build a joint probability model over documents and words, defined as the mixture. pLSA introduces a conditional independence assumption, namely that the occurrence of a word w is independent of the video document or frame d it belongs to, given a topic z. The model is defined by the probability of a document P(d), the conditional probabilities P(w|z) which represent the probability of observing the word w given the topic z, and by the document-specific conditional multinomial probabilities P(z|d). The topic model decomposes the conditional probabilities of words in a document P(w|d) as a convex combination of the topic specific word distributions P(w|z), where the weights are given by the distribution of topics P(z|d) in the document. Although pLSA is a non-fully generative model, its tractable likelihood maximization makes it an interesting alternative to fully generative models like LDA with comparative performance. The parameters of the model are estimated using the maximum likelihood principle.

ANOMALY DETECTION



The estimation of the topic distribution P(z|d) of a given frame is obtained by optimizing log-likelihood function. We use this log-likelihood measure to consider if a frame is normal or abnormal. If the behaviors happening within the frame corresponds to those observed in the training dataset, then we should be able to find a suitable topic distribution explaining the bag-of-word representation of the frame. Thus, normal frames will generally provide high log-likelihood. On the other hand, if an abnormal activity is going on, none of the learned topic will able to explain the observed words of that behavior, resulting in a low likelihood fit. Log-likelihood function is defined as follows:

$$L_d^u(P(z|d)) = \sum_w \mathrm{n}(d,w) \log \left(\sum_z P(z|d)P(w|z) \right)$$

However, it suffers from a severe drawback. It is not normalized and thus, whatever the quality of the fit, the measure is highly correlated with the document size. To solve this issue, we exploit the average log-likelihood of each word, by dividing n(d, w) by the number of words. The normalized log-likelihood measure is defined as follows:

$$\begin{aligned} L_d^{nl}(P(z|d)) &= \sum_w \frac{\mathrm{n}(d,w)}{\mathrm{n}_d} \log \sum_z P(z|d)P(w|z) \\ &= \sum_w P_o(w|d) \log P_c(w|d) \qquad (6) \end{aligned}$$

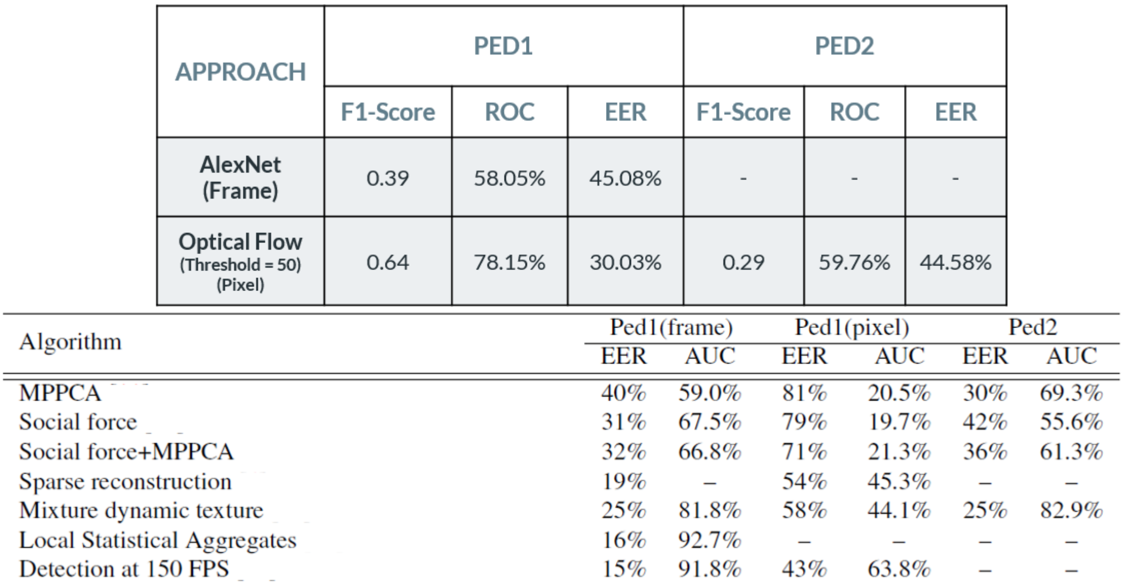
where $P_o(w|d) = n(d,w)/n_d$ is called the objective distribution as it is measured directly from the test document, and $P_c(w|d)$ is called the constrained distribution as it lies in the constrained simplex spanned by the topic distribution P(w|z). This approach is inspired by [9].

EXPERIMENTS

We formed clusters on a smaller subset of our dataset and learned a language model over it. We were able to achieve a precision of **0.55** and recall of **0.32**.

EVALUATION

We compare our results with some previously proposed unsupervised approaches:



APPROACH	PED1			PED2		
	F1-Score	ROC	EER	F1-Score	ROC	EER
AlexNet (Frame)	0.39	58.05%	45.08%	-	-	-
Optical Flow (Threshold = 50) (Pixel)	0.64	78.15%	30.03%	0.29	59.76%	44.58%

Algorithm	Ped1(frame)		Ped1(pixel)		Ped2	
	EER	AUC	EER	AUC	EER	AUC
MPPCA	40%	59.0%	81%	20.5%	30%	69.3%
Social force	31%	67.5%	79%	19.7%	42%	55.6%
Social force+MPPCA	32%	66.8%	71%	21.3%	36%	61.3%
Sparse reconstruction	19%	–	54%	45.3%	–	–
Mixture dynamic texture	25%	81.8%	58%	44.1%	25%	82.9%
Local Statistical Aggregates	16%	92.7%	–	–	–	–
Detection at 150 FPS	15%	91.8%	43%	63.8%	–	–

Figure 8 Comparison with unsupervised approaches

We find that our supervised approaches perform much better than existing unsupervised approaches both at pixel and frame level.

Conclusion & Future Work

In this project we explored 4 supervised approaches of anomaly detection in videos:

- **Optical Flow**
- Feature Extraction from **AlexNet**
- **Time Series Analysis** over Video
- Utilizing **Topic Modelling**

Our approaches include both frame level and pixel level anomaly detection. We found that Supervised Approaches outperform unsupervised ones.

In future, we would like exploit the proposed 4 techniques in different contexts. In particular, we would like to vary our dataset and see how these approaches perform. In addition to this, we aim to combine these approaches to form a much robust anomaly detection system.

REFERENCES

1. Rousseeuw, P. and Leroy, A.: 1996, Robust Regression and Outlier Detection. John Wiley & Sons., 3 edition
2. Torr, P. H. S. and Murray, D. W.: 1993, 'Outlier Detection and Motion Segmentation'. In: Proceedings of SPIE
3. Fawcett, T. and Provost, F. J.: 1999, 'Activity Monitoring: Noticing Interesting Changes in Behavior'. In: Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 53–62
4. Japkowicz, N., Myers, C., and Gluck M. A.: 1995, 'A Novelty Detection Approach to Classification'. In: Proceedings of the 14th International Conference on Artificial Intelligence (IJCAI-95). pp. 518–523
5. http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm
6. Saligrama, Venkatesh, and Zhu Chen. "Video anomaly detection based on local statistical aggregates." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.
7. https://github.com/BVLC/caffe/tree/master/models/bvlc_reference_caffenet
8. T. Hofmann. Unsupervised learning by probability latent semantic analysis. Machine Learning, 42:177–196, 2001
9. Varadarajan, Jagannadan, and Jean-Marc Odobez. "Topic models for scene analysis and abnormality detection." *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE, 2009.