Anomalous Human Behavior Detection: An adaptive approach

Coen van Leeuwen[a], Arvid Halma[b] and Klamer Schutte[a]

[a]TNO Intelligent Imaging, Oude Waalsdorperweg 63, The Hague, the Netherlands
[b]Research Kitchen, Rotterdamseweg 183C, Delft, the Netherlands

ABSTRACT

Detection of anomalies (outliers or abnormal instances) is an important element in a range of applications such as fault, fraud, suspicious behavior detection and knowledge discovery. In this article we propose a new method for anomaly detection and performed tested its ability to detect anomalous behavior in videos from DARPA's Mind's Eye program, containing a variety of human activities. In this semi-unsupervised task a set of normal instances is provided for training, after which unknown abnormal behavior has to be detected in a test set. The features extracted from the video data have high dimensionality, are sparse and inhomogeneously distributed in the feature space making it a challenging task. Given these characteristics a distance-based method is preferred, but choosing a threshold to classify instances as (ab)normal is non-trivial. Our novel aproach, the Adaptive Outlier Distance (AOD) is able to detect outliers in these conditions based on local distance ratios. The underlying assumption is that the local maximum distance between labeled examples is a good indicator of the variation in that neighborhood, and therefore a local threshold will result in more robust outlier detection. We compare our method to existing state-of-art methods such as the Local Outlier Factor (LOF) and the Local Distance-based Outlier Factor (LDOF). The results of the experiments show that our novel approach improves the quality of the anomaly detection.

Keywords: Anomaly Detection, One-class Classification, Outlier Detection, Pattern Recognition, Video Analysis

1. INTRODUCTION

As part of DARPA's Mind's Eye program we developed a system for detecting anomalous behavior in videos containing a variety of human activities. In such tasks one typically only has training data available for the case with normal, insuspicious behavior; representative footage of abnormal behavior is too rare or even entirely unavailable.

The system's input is a 48-dimensional feature vector of confidences of detected verbs. A word cloud of these verbs is shown in Figure 1. By setting a threshold on the distance of the queried state to known 'normal' instances, an outlier is defined: when the distance is smaller than the threshold it is similar to the observed normal behavior, otherwise it is too deviant and considered an anomaly.

Choosing a threshold in this high-dimensional space is difficult. Even though the idea of having a fixed distance threshold is intuitive for human operators, a single threshold implies a homogeneous spread of known instances. In this paper we therefore explore adaptive thresholds to find anomalous behavior in a more robust manner with a more meaningful parameter. In section 1 we will state the problem and introduce the main idea of our new method: the Adaptive Outlier Detection (AOD) algorithm.

We shall then discuss existing methods in section 2, followed by a more detailed description of the new algorithm in section 3. In section 4 we will first show results of a synthetic experiment, after which we evaluate behavior detected from video data in order to find anomalous actions. Finally in section 5 we will discuss what we can learn from this, and consider future work.

Further author information:
C.J. van Leeuwen: E-mail: coen.vanleeuwen@tno.nl, Telephone: +31 (0)88 86 63 113
Arvid Halma: E-mail: arvid@researchkitchen.net, Telephone: +31 (0)15 268 25 87

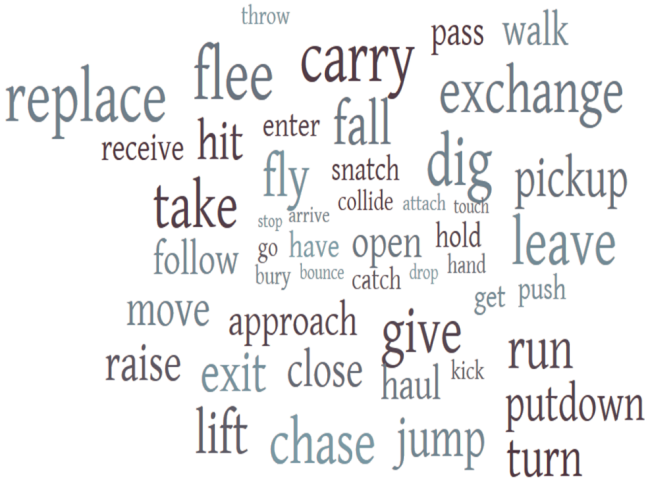


Figure 1. A word cloud of the used verbs in the DARPA Mind's eye program

1.1 Task Description

In our application we have a video data set of various settings where humans exhibit a wide range of behaviors. In total, 48 actions are distinguished that are of interest. To give an idea, some verbs that are included are: *approach*, *bury*, *catch*, *drop*, *exchange*, *walk* (for a complete list see Figure 1). Dedicated classifiers attempt to recognize these actions in certain parts of a scene at every moment.[1–3] The results of these classifiers on all verbs are represented as verb probabilities, and form the input for the anomaly detection system.

1.2 Types of anomaly detection tasks

There are basically three types of anomaly recognition tasks[4] which are shown in Table 1:

The first type is analogous to the unsupervised clustering task where the model has no prior knowledge of the data. The system simply receives a data set as input and the goal is to find instances that are in some way anomalous. This type of task is typically done on static data where the main interest is to find interesting items amongst the bulk of "normal" data.

The second type of anomaly detection is analogous to supervised classification. The system is initially trained using both "normal" and "abnormal" instances, and it simply learns to classify new instances accordingly. This approach is very useful when the type of anomalies is known and abundant before they occur. They will however not be able to handle completely new types of anomalies that are not present in the training phase.

Finally, the third approach of anomaly detection is analogous to a semi-supervised classification task. In this task the system is initially offered only data labeled as "normal" in a training phase. The system will need to create a model of this normality and will determine some boundaries within which data is assumed to be normal. After that unclassified instances need to be labeled "normal" or "abnormal" appropriately. The assumption made for this type of task is that the full spectrum of "normal" data is observed before the testing starts.

This third type of anomaly detection is most relevant to the rest of this paper, as it is most like the task we are given in the video behavior analysis. In any type of the anomaly detection task we would like to derive some weight or confidence level that indicates to which extent we can rely on the given classification.

1.3 Evaluating anomaly detection methods

The performance of an anomaly detection system can be evaluated in different ways. All measures are based on the number of true positives (TP: correct detection), true negatives (TN: correct silence), false positives (FP: false alarms) and false negatives (FN: missed detection) when comparing returned output to a ground-truth data set. In our case, we find it important not to miss a true anomaly, but also to make not too many false alarms. Because in most practical cases the amount of anomalies greatly outnumber the number of normal instances, we want to balance the types of errors. The Matthews correlation coefficient incorporates this desire:

Table 1. Of the three different types of anomalies, in this paper primarily semi-supervised anomaly detection will be considered

		Abnormal instances in Training data	
		YES	NO
Labeled Data	YES	Classification	**Semi-supervised Anomaly Detection**
	NO	Knowledge Discovery	N/A

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{1}$$

Therefore it is our primary indication for evaluating the selected methods.

2. RELATED WORK

There are many different methods to determine anomalies from data sets,[4] and we shall not list them all in this paper. However there are some methods that are can be clustered into a group of distance based classifiers that base their output on a distances in the feature space. The method that we propose in this article also belongs to this group, since we will use local neighborhood distances to determine a point's anomaly likelihood without assuming an overall distribution model.[5]

Other methods that somehow fit a mixture of parametric models (such as Parzen density estimators[6,7] or expectation maximization for determining outlier probabilities[8] make fundamentally different assumptions on the occurrence of outliers. Where they assume outliers can be discriminated in areas that are sparsely covered by the normal instances, distance based methods do the opposite. Distance based methods are based on the assumption that sparsely covered subspaces containing normal instances support a larger area in which unseen normal instances can occur. That is why such methods are not included in our comparison.

The *Local Outlier Factor* (LOF)[9] is a measure of how much of an outlier a query point is compared to a data set. In order to compute this value, a measure called the *reachability density* for every point needs to be defined, which is the local density of k nearest neighbors up to a specific bound. The LOF is then determined by comparing the reachability density from one point to that of its k nearest neighbors. The authors suggest using a relatively large number of comparison points (large k) in order to get reliable results.

A similar measure is the *Local distance-based outlier factor* (LDOF).[10] This metric is much more straightforward in that it does not compare local density estimations, but instead compare the average distance to its k nearest neighbors to the average inner distance of those k neighbors.

The Local Correlation Integral (LOCI)[11] is another method that uses local distances. It compares the distribution of pair-wise distances in different ranges around a query point. This provides a local density estimation around a point, as well as a local density deviation on which they base their anomaly classification.

3. METHODS

3.1 Fixed or dynamic threshold

In order to make a classification of a query point that is being compared to a reference set of labeled normal data, we implemented a method on very similar assumptions to the nearest neighbor classification method. Nearest neighbor search is not directly suitable for anomaly detection, since a mapping from nearest neighbor distances to

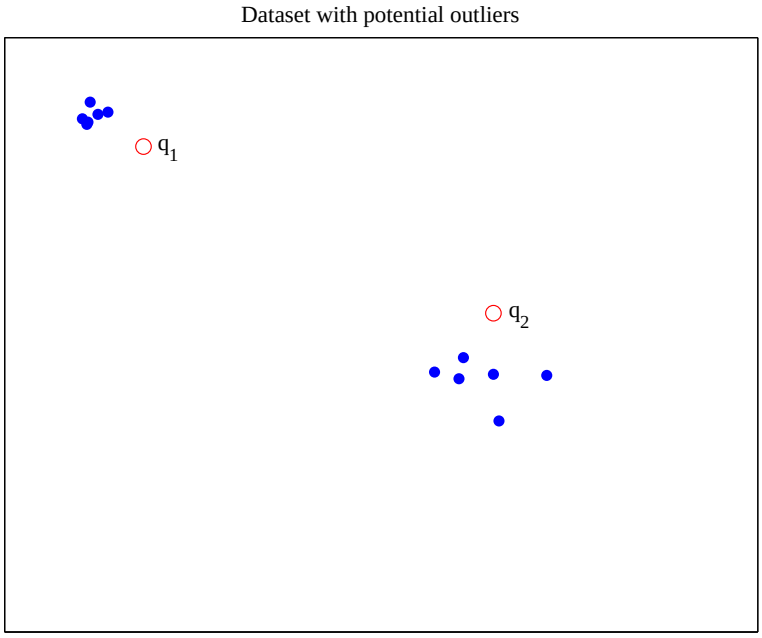


Figure 2. A random data set with two clusters of different densities. Two query points at similar distance to the clusters may be classified differently

anomaly classification has yet to be made. The simplest method to do this is to threshold the Euclidean distance measure with a fixed predetermined threshold. Choosing the threshold is not part of the responsibilities of the algorithm, and may be difficult to choose in practice, since most real world data is unlikely to be distributed homogeneously.[5] A difference in local densities may suggest different thresholds, as is depicted in Figure 2.

Query point q_1 is a potential outlier located near the lower left cluster, where q_2 is placed near a less dense cluster and therefore less likely to be a relevant outlier. For both points the distances to their respective nearest neighbor are equal. In this case a fixed threshold cannot discriminate the two query points.

Choosing a different bilinear distance metric (such as any L-distance or in particular the Mahalanobis distance) can weigh points in such a way that all dimensions contribute equally to the outcome. It does not help to solve the difficulties with a fixed threshold entirely, as it linearly deforms the entire space and therefore is not able cancel out the effect of clusters within the data set. If the feature space in Figure 2 would be transformed using Mahalanobis distance, it would even suggest the opposite from our intuition since q_2 would be considered the outlier.

Methods using dynamic thresholds on distance metrics are able to correctly identify just these type of problems where outliers in a data set depend on the distance to a nearby cluster, and the density of that cluster. For example, a point at a medium distance to a relatively dense cluster is an outlier, whereas a point near a sparse cluster exhibiting similar distances to its neighbors is not.[12]

3.2 Adaptive Outlier Detection

Our definition of the outlier distance is depicted in Figure 3, and is defined as follows:

Suppose we have some reference data set R and a query point $q \notin R$. We then define $R_{kq} \subset R$, which consists of the collection of the k nearest neighbors in R to q. Correspondingly we have D_{kq} which is the set of distances of R_{kq} to q. Also we have D_{mr} which is the collection of the distances from all points $r \in R_{kq}$ to their m nearest neighbors in R (so D_{mr} is a collection of $k * m$ distances). We can then define the adaptive outlier distance of q as follows:

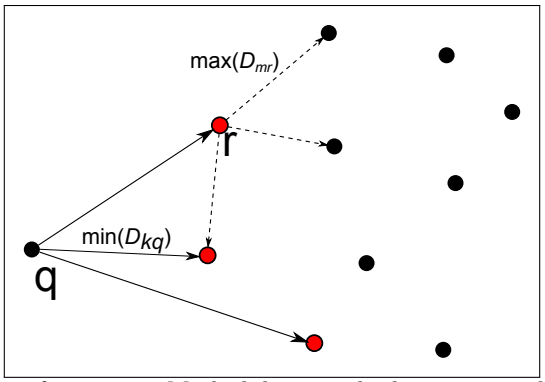


Figure 3. A point q is compared to a reference set. Marked dots are the k nearest neighbors, with corresponding distances (D_{kq}) as solid lines. Of one neighboring point r its m nearest neighbor distances (D_{mr}) are shown as dashed lines

$$AOD_q = \frac{\min\left(D_{kq}\right)}{\max\left(D_{mr}\right) + \epsilon} \tag{2}$$

$$\text{anomaly} = \begin{cases} true & \text{if } AOD_q > \alpha \\ false & \text{if } AOD_q \leq \alpha \end{cases} \tag{3}$$

What this means is that the outlier distance of a point q is the ratio of the distance to its nearest neighbor and the maximum distance of any neighbor's m^{th} next neighbor. This ratio of the first and second hop distance from any query point q indicates wether the query point can be considered normal according to the variation in R. The factor ϵ is a very small number to prevent division by zero. The value of the AOD ratio is hardly influenced by this value given non-zero distances, and for $\max\left(D_{mr}\right) = 0$ it ensures the outlier distance goes toward infinity which is conform to our design.

For the simple case where $k = 1$ and $m = 1$ this means simply comparing the distance of the nearest neighbor to the distance of that neighbor to *its* nearest neighbor. The method then determines whether a point is an outlier if the value from equation 2 is larger than some threshold α, and if it is, it is considered an outlier. In the implementation we leave the choice for m open, but in this article we from here on assume $m = k$.

When comparing the above approach to LDOF, there are some similarities, but there is one significant difference. Once you assume that all given normal instances are labeled correctly, any point can support large areas to be classified as 'normal' if a large spread occurs in the normal data. For our method we then hypothesize to use the maximum distance ratio found in the k neighbors considered in contrast to the average distance. This is based on the assumption that a query point only has to be explained as being 'normal' by one of its neighbors, not by a weighted voting of k neighbors.

In this article we focus on the different distance comparing methods, not what metric we define our distances in. Choosing a distance metric may have impact on the performance.[13] In this article we will only look at methods using a Euclidean distance metric (L_2 norm) because of its intuitive notion of distance and ease of use. We assume that when comparing the different approaches, a single distance metric will not favor one method over another. This same assumption is made in other studies.[11]

3.3 Unsupervised AOD

The algorithm described so far classifies a query point based on a set of "normal" reference points in a semi-supervised classification task. However, as described in 1.2 there is also the unsupervised task in which we simply want to find out anomalous points in a set of data points. For this task we simply do an iterated call to the regular AOD algorithm with every point $q \in R$ and take $R' = R \setminus \{q\}$. Then for each member $q \in R$ we can determine if that would be an outlier compared to R'.

Repeating this until there are no more points considered as outlier we obtain a new set $R^* \subset R$ of points that are "normal". For the unsupervised task we are now done, each point that is not in R^* is considered anomalous.

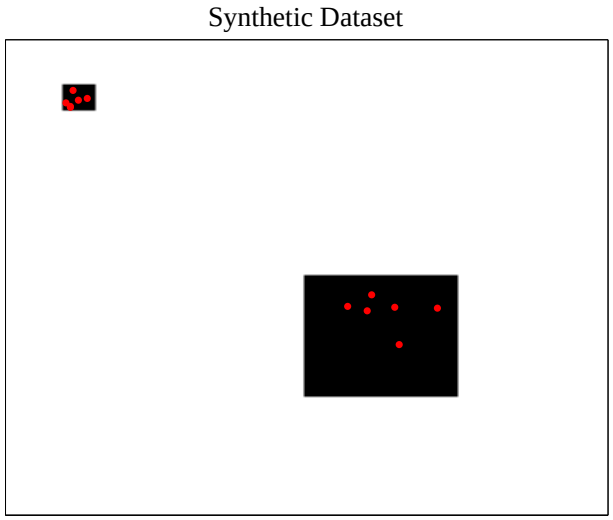


Figure 4. The ground-truth p.d.f. with all possible inliers in black and in white the outliers. The 40 uniformly drawn samples are shown as dots.

For the supervised task however, we now have the possibility to determine if a query point q is anomalous when comparing it to R^*: this is the unsupervised variant of the AOD algorithm.

3.4 Implementation

The proposed method for anomaly detection is based on k nearest neighbor (k-NN) classification, in the sense that new instances are compared locally when doing classification. There exist multiple implementations of k-nearest neighbor search, where the point data is organized as lists (linear search or as priority queue/binary heap) or as trees (such as k-d trees, quad trees, octtrees). Depending on the data set, one data structure may be preferred over the other. For example, with relative few high-dimensional points lists may be preferred over trees that come with overhead in dividing subspaces. In other cases the partitioning performed by trees may reduce k-nearest neighbor lookup significantly. For our test purposes we implemented a simple point list.

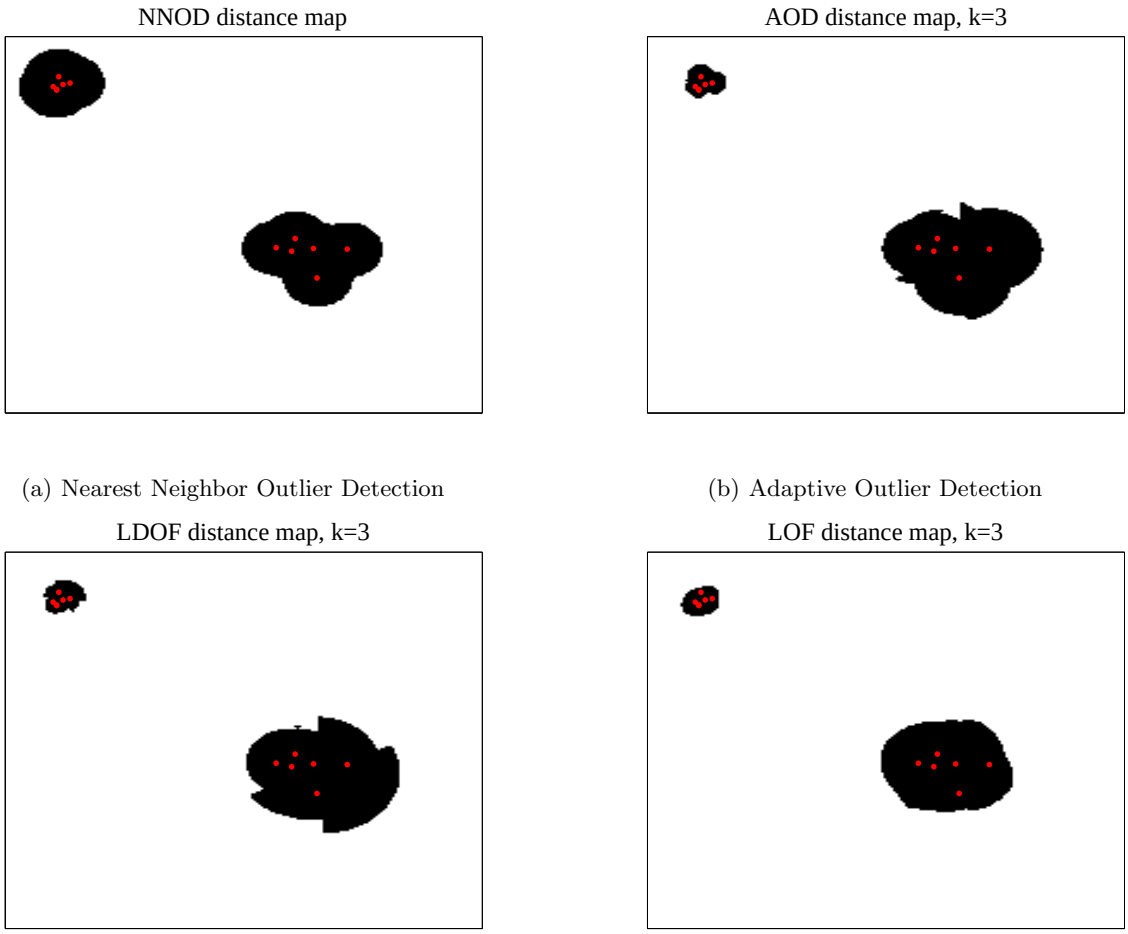
For this implementation there is not a significant training step (other than adding known instances to the data set). As a consequence of this choice, we inherit some advantages directly: it is relatively simple to implement and non-parametric (no a priori distribution is assumed). Disadvantages include the need to store all points instead of a more memory efficient model, and that classification may be more computationally expensive than methods that preprocess during training.

4. EXPERIMENTS

4.1 Synthetic experiments

To see how the various algorithms perform, their output is evaluated on some artificial data set. The ground truth of this artificial data set is formed by two discrete bounded 2D areas of different size, where each element has probability 1 to be normal, for all points outside the areas the probability is zero. The shape of this area is shown in Figure 4. The input for the algorithm is a sampled set of n points drawn uniformly from each area: labeled normal instances. The classified output of an algorithm can then be compared the actual inlier/outlier ground truth. The bigger the overlap between output and ground truth p.d.f. is, the better that algorithm performed.

In the experiment the value of α is determined on a training set of samples from the complete search space, after which the MCC score is determined on an evaluation set of different random samples. In Table 2 the combined results are shown for 20 experiments using different random seeds. As can be seen in this case the



(a) Nearest Neighbor Outlier Detection

(b) Adaptive Outlier Detection

(c) Local Distance-based Outlier Factor

(d) Local Outlier Factor

Figure 5. In black the areas are shown that would be considered normal based on the different anomaly detection algorithms when applied to the samples from Figure 4

adaptive thresholding improve the anomaly detection perfomance. However in this case, the LDOF algorithm outperforms the AOD algorithm marginally. In this experiment the used parameters were $n = 6$ and $k = 3$.

Table 2. MCC scores of the different algorithms on the synthetic task.

Algorithm	MCC
NNOD	0.564
AOD	0.646
LDOF	0.693
LOF	0.578

4.2 Behavioral anomalies

In the second experiment we compare different anomaly detection methods on how well they are able to detect anomalous behavior. The actual action recognition is not in the scope of this paper, instead we used a data set which had already been classified.[2] In this data set of over 3500 clips, each clip was given a set of 48 probabilities corresponding to 48 verbs. We also had access to human annotated ground truth of the majority of these clips.

A series of experiments were performed on this data set. Each experiment is set up as follows:

1. One single verb is chosen which is selected as an *anomalous* activity.
2. A training set is created by randomly choosing 150 clips where there is no presence in the ground truth for the selected verb.
3. Next, a test set is created by randomly choosing 475 clips where there is still no presence of the anomalous verb, plus 10 clips where the anomalous verb *is* present.
4. An experiment is performed where each clip in the test set is compared to the training set, and a decision is made wether the clip is anomalous or not.

The experiment is repeated using every verb as being anomalous. Every method then has to classify every clip in the test set, and their scores are evaluated using the MCC score. We expect that in each experiment, the 10 clips with the added verb are found to be anomalous. Based on the training set, these anomalous verbs should have a lower probability.

It turned out that when faced a 48 dimensional problem, all of the methods performed very poorly, so that none of them performed better than a random classifier. Therefore we decided to make the problem more feasible be reducing the amount of dimensions to 10. This means that all predictions of the remaining 38 verbs are ignored; they are not selected as being anomalous, nor are they used to predict anomalousness. Which verbs are used and which ones are ignored was chosen randomly, and many different permutations of used verbs have been tested.

For each selection of verbs, the experiment is repeated 10 times so that each verb is considered anomalous once. The results are then averaged to get results for the current selected set of verbs. First we will analyse one such specific set using an ROC curve, and then look at the influence of the choice of verbs by comparing the highest MCC for any choice of parameters (k and α).

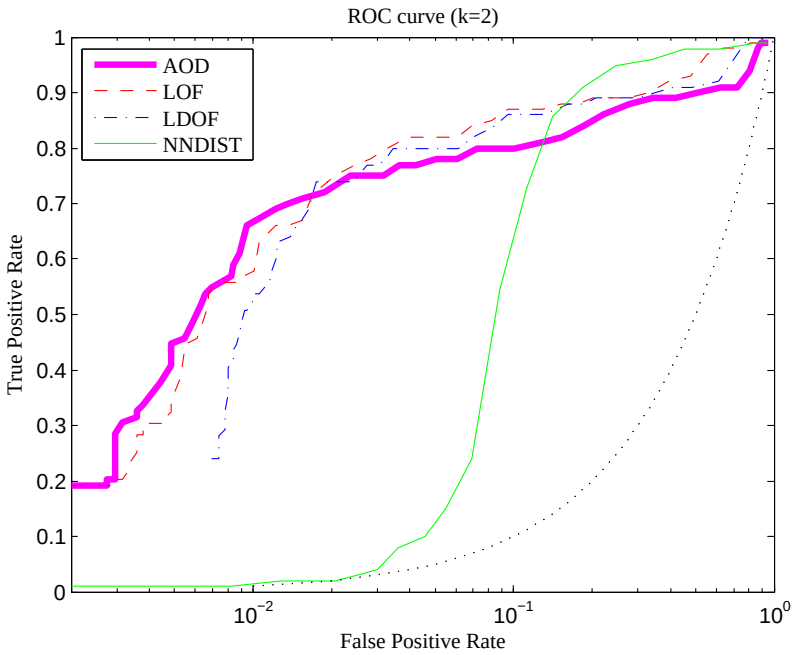


Figure 6. The ROC curves of the behavioral anomaly experiment for different methods. The dashed line represents the expected result of a weighted random classifier.

In Figure 6 an ROC curve with the x-axis is shown in log scale, showing where the AOD algorithm outperforms the others. Because by definition of the experiment there are much more inliers then outliers, the false alarm rate plays a significantly large role in calculating the MCC score. Because AOD actually outputs less false alarms

then other methods, its maximum MCC value is higher then LOF's. This is because *more* new data is covered by the normal field by AOD then by any other method. This means that it would be especially useful when false alarms are generally expensive, or when there are simply very few anomalies in a large amount of normal instances. The numerical values of the maximum MCC along the curves can be seen in the second column in Table 3.

In Figure 6 the results are shown for an experiment in which one combination of 10 verbs are shown (particularly they are: *bury*, *catch*, *chase*, *exit*, *follow*, *give*, *open*, *pick up*, *push* and *put down*). When choosing different sets of verbs a different anomaly/normal segmentation is possible. In Figure 7 we show the MCC scores for almost 3000 different random permutations of verb combinations. In this figure it can be seen that the choice of the verbs is a crucial one, since the MCCs vary between 0.1 and 0.9, however we do see that AOD and LDOF outperform LOF in most cases, and LOF in often outperforms the static Nearest Neighbor method.

In order to compare the different methods a range of parameter settings was used, and the best found combination was selected to compare. Specifically for α values from 0.05 through 5 were used with increments of 0.05, and for k the values $1, 2, 3, 5, 7, 11, 15$ were used.

The particular choice of verbs that was chosen for the experiment in Figure 6 is the set of verbs where the sum of the four MCCs is maximal. This is a set of verbs where every method performs relatively well, but for every method there exists at least one set where it performs better. In Table 3 the MCC scores are shown for this set where every method performs well, as well as the best individual MCC score we found amongst all verb permutations.

Table 3. MCC scores of the different algorithms on the behavioral anomaly task for optimally chosen thresholds for each method.

Algorithm	Max sum MCC	Max individual MCC
AOD	0.861	0.930 ($k = 3$)
LOF	0.747	0.794 ($k = 1$)
LDOF	0.861	0.921 ($k = 3$)
NNdist	0.156	0.373

5. CONCLUSION AND FUTURE WORK

In this paper we show that we can improve the performance of outlier detection using an adaptive threshold on distance based. A new method for dynamic outlier detection is proposed, and compared to similar existing solutions. Our new method does very little assumptions about the distribution of the data, except that points that are similar are close together in some feature space.

In the experiments with the synthetic data set we saw an improvement with the adaptive methods, compared to a fixed distance nearest neighbor algorithm. We did see however that the LDOF method scored marginally better than the AOD method. In the synthetic experiments our assumptions about the maximum distance being representative for the spread in the normal data did not lead to a better classification.

In the human behavioral experiments we have also shown that the adaptivity allows for a great gain in performance. Both LDOF and AOD largely outperform the static method, the LOF algorithm also performed better, but by a smaller margin. In this experiment the AOD algorithm outperformed the LDOF method, so in this case the maximum distance between normal instance was a more representative measure for the normal cluster density then the average distance.

It seems that the proposed method is an interesting new method for automatically finding outliers. Compared to other methods it performs very well in both low and high dimensional data. The method appears specially well in keeping low false alarm rates, making it an good choice for many tasks that involve few anomalies with high information level. However there are some circumstances in which it performs better than LDOF and some in which it does not. This has to do with the distribution of normal instances in the feature space, but in order to make it clear what the exact properties of those distributions are, more research is needed.

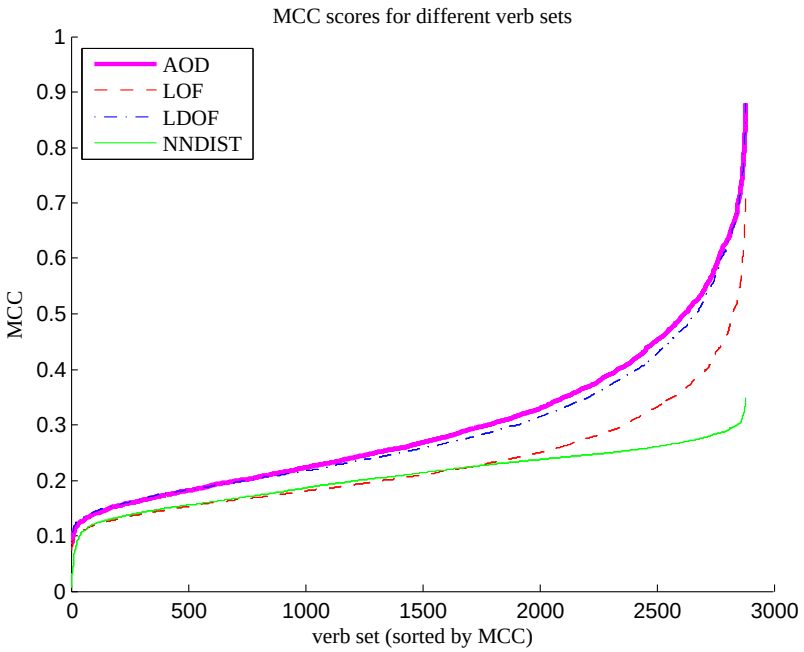


Figure 7. The maximum MCC score of the different methods are shown for different verb sets. The scores are sorted for each method to compare the score distributions, not individual verb sets.

In the experiments of the human behavior anomalies we have shown that different sets of verbs perform better than others. This is probably greatly influenced by the quality of the classification of the verb probabilities (the input). However in this article, no extensive analysis is done on what combinations of verbs perform better, and why this is the case. This would be a very interesting study for future research. It could be true that the well-performing combinations are interesting to select for human anomaly detection in general, or the exact opposite: they could in fact all be so close together, that they are separable in this context, but are completely irrelevant for the *ignored* dimensions. Either finding would be of interest for the verb classification task.

ACKNOWLEDGEMENTS

This work is supported by DARPA (Minds Eye program). The content of the information does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred. The authors acknowledge the Cortex team for their contributions.

REFERENCES

[1] Burghouts, G., Bouma, H., de Hollander, R., van den Broek, S., and Schutte, K., "Recognition of 48 human behaviors from video," in [*Int. Symp. Optronics in Defense and Security, OPTRO*], (2012).

[2] Bouma, H., Raaijmakers, S., Halma, A., and Wedemeijer, H., "Anomaly detection for internet surveillance," in [*SPIE Defense, Security, and Sensing*], 840807–840807, International Society for Optics and Photonics (2012).

[3] Bouma, H., Burghouts, G., Penning, L., Hanckmann, P., ten Hove, J., Korzec, S., Kruithof, M., Landsmeer, S., van Leeuwen, C., van den Broek, S., Halma, A., den Hollander, R., and Schutte, K., "Recognition and localization of relevant human behavior in videos," in [*Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*], **8711** (2013).

[4] Hodge, V. and Austin, J., "A survey of outlier detection methodologies," *Artificial Intelligence Review* **22**(2), 85–126 (2004).

[5] Burghouts, G. J., den Hollander, R., Schutte, K., Marck, J.-W., Landsmeer, S., and den Breejen, E., "Increasing the security at vital infrastructures: automated detection of deviant behaviors," in [*SPIE Defense, Security, and Sensing*], 80190C–80190C, International Society for Optics and Photonics (2011).

[6] Bishop, C., "Novelty detection and neural network validation," in [*Vision, Image and Signal Processing, IEE Proceedings-*], **141**, 217–222, IET (1994).

[7] Tax, D. and Duin, R., "Outlier detection using classifier instability," *Advances in Pattern Recognition* , 593–601 (1998).

[8] Stein, D., Beaven, S., Hoff, L., Winter, E., Schaum, A., and Stocker, A., "Anomaly detection from hyperspectral imagery," *Signal Processing Magazine, IEEE* **19**(1), 58–69 (2002).

[9] Breunig, M., Kriegel, H., Ng, R., Sander, J., et al., "LOF: identifying density-based local outliers," *Sigmod Record* **29**(2), 93–104 (2000).

[10] Zhang, K., Hutter, M., and Jin, H., "A new local distance-based outlier detection approach for scattered real-world data," *Advances in Knowledge Discovery and Data Mining* **1**, 813–822 (2009).

[11] Papadimitriou, S., Kitagawa, H., Gibbons, P., and Faloutsos, C., "LOCI: Fast outlier detection using the local correlation integral," in [*Data Engineering, 2003. Proceedings. 19th International Conference on*], 315–326, Ieee (2003).

[12] Kriegel, H., Kröger, P., Schubert, E., and Zimek, A., "Outlier detection in axis-parallel subspaces of high dimensional data," *Advances in Knowledge Discovery and Data Mining* , 831–838 (2009).

[13] Aggarwal, C., Hinneburg, A., and Keim, D., "On the surprising behavior of distance metrics in high dimensional space," *Database TheoryICDT 2001* , 420–434 (2001).