A Survey of Single-Scene Video Anomaly Detection

Bharathkumar Ramachandra, Member, IEEE, Michael J. Jones, Senior Member, IEEE, and Ranga Raju Vatsavai, Member, IEEE

Abstract—This survey article summarizes research trends on the topic of anomaly detection in video feeds of a single scene. We discuss the various problem formulations, publicly available datasets and evaluation criteria. We categorize and situate past research into an intuitive taxonomy. Finally, we also provide best practices and suggest some possible directions for future research.

Index Terms—video anomaly detection, abnormal event detection, surveillance

1 INTRODUCTION

Video anomaly detection is the task of localizing anomalies in space and/or time in a video, where anomalies are simply activities that are out of the ordinary. Anomalies are also referred to as abnormalities, novelties, and outliers among other similar terms. Examples range from unattended bags at airports, to people falling down, to a person loitering outside a building. We follow the definition provided in [1].

Definition 1 Video anomalies can be thought of as the occurrence of unusual appearance or motion attributes or the occurrence of usual appearance or motion attributes in unusual locations or times.

One implication of this definition that is not immediately obvious is that video anomalies are scene-dependent. This means that activity that is anomalous in one scene may be normal in another. For example, in one scene, riding a bicycle along a bike path is normal, while in another, riding a bicycle down a similar looking pedestrian sidewalk is anomalous. Normal video (that is, video not containing any anomalies) is thus needed for model training to express the variety of normal activities that may occur in a particular scene. Since it is unrealistic to collect video for all possible anomalous events for training and expensive to collect even a few anomalous events, a common assumption is that training data consists of only normal activities which is relatively easy to obtain.

The currently available datasets for single-scene video anomaly detection (UCSD Ped1 & Ped2 [2], CUHK Avenue [3], Subway [4], UMN [5], and Street Scene [6]) are also static-camera datasets, but the camera being static is not necessary. In fact, the CUHK Avenue [3] and Street Scene [6] datasets do have some minor camera motion. One can imagine a model for a single scene being able to handle camera motion when the majority of each frame overlaps with neighboring frames (as would occur with a pan-tilt-zoom surveillance camera) by keeping track of global location in each frame. Such a formulation would still be considered single-scene. There are currently no benchmark datasets available or algorithms proposed for this single-scene moving-camera version of the problem, but it is a fertile area for future research.

Figure 1 shows an overview of typical algorithms for single-scene video anomaly detection. First, during a training phase, a model of normal activity is learned from the features computed from one or more videos of a scene which do not contain anomalies. Then in the detection phase, new video from the same scene is given from which the same types of features are computed. The features along with the model are used to assign anomaly scores to each voxel of the input video. Anomaly scores are then thresholded to yield spatio-temporal binary masks of the anomalies detected.

1.1 Other formulations of the problem

It is important to point out that many papers on video anomaly detection have addressed different formulations of the problem than the single-scene formulation on which this survey focuses. One such formulation is multi-scene video anomaly detection ([7], [8], [9], [10], [11]). In this version of the problem, normal video from many different, unrelated scenes is given for building a single model of normality. The goal in this case is to learn the normality in a variety of appearances and activities that occur anywhere in any of the videos. For example, each of the normal videos may simply show people walking. Any other activity in testing video (such as people fighting, people riding bikes or golf carts) should be detected as anomalous. Building a single model across multiple scenes may not make sense unless the scenes are “consistent” in the sense that an anomaly in one scene is also an anomaly in another. For example, several cameras in different locations of a single large store mainly monitoring for theft would be “consistent”. Alternately, consider the...
1. Training Phase

![Diagram of Normal training videos of a scene]

2. Detection Phase

![Diagram of Testing video of same scene]

Fig. 1. Overview of single-scene video anomaly detection. Typical algorithms include a model building phase in which a model of normal activity is learned from one or more videos of a scene followed by a detection phase in which anomalies are detected in video from the same scene.

case of two cameras in which the scene from the first camera contains a restricted grassy area that people are not allowed to walk through (and thus walking in that area is anomalous), while the second camera has grassy areas that are not restricted zones. A single model trained to detect anomalies in both scenes would not be able to detect the anomalous grass-walking activity in the first scene since it will have learned from the second scene that walking on grass is normal. This example illustrates that location-dependent anomalies are compatible with single-scene video anomaly detection but not with multi-scene video anomaly detection. Location-dependent anomalies (such as jaywalking, riding a bike on a pedestrian sidewalk, driving a car in the wrong direction and etcetera) which involve normal activity occurring in unusual locations are commonplace for single-scene video anomaly detection, but are not compatible with a multi-scene formulation due to the lack of correspondence among locations in different scenes. Naturally, methods that perform in a location-dependent fashion perform better on the single-scene datasets when compared to multi-scene datasets and vice versa.

Another alternate formulation for video anomaly detection that has been used in a number of papers ([12], [13], [14], [15]) is training-free video anomaly detection. In this formulation, no normal training video is provided and the task is to either detect changes in the testing video or else to detect the most unusual segments of the testing video as proxies for anomalousness. Detecting the most unusual segments of testing video is analogous to discord detection in time series analysis [15]. While these formulations of the problem are also useful, they are significantly different from single-scene video anomaly detection and require different datasets and ground truth annotations.

Many existing research papers do not clearly distinguish which problem formulation they are using. This leads to ambiguities and confusion about what datasets should be tested on and which methods should be compared against. It also leads to differences in understanding the performance of different methods. We think it is important to make clear the problem formulation being used in any paper on video anomaly detection. In this survey, we have chosen to focus on the single-scene video anomaly detection formulation because it encompasses a very common scenario and has many practical applications, such as in surveillance, security, factory automation (monitoring the activity of workshop floors), video search and video summarization.

1.2 Types of Video Anomalies

Here we attempt to provide a non-exhaustive list of what we think are the most commonly occurring video anomalies; a specific application may warrant the declaration of other types of anomalies.

1.2.1 Appearance-only Anomalies

These anomalies can be thought of as unusual object appearance in a scene. Examples include bicyclists on a pedestrian walkway, or a large boulder on a road. Detecting these anomalies only requires inspecting a local region of a single frame of video.

1.2.2 Short-term Motion-only Anomalies

These anomalies can be thought of as unusual object motion in a scene. Examples include a person running in a library, or loitering around foreign embassies. Detecting these anomalies usually only require inspecting a local region of the video over a short period of time. Appearance-only and short-term motion-only anomalies can be further called local anomalies because they possess this additional property.
1.2.3 Long-term Trajectory Anomalies

These anomalies can be thought of as unusual object trajectory in a scene. Examples include persons walking in a zig-zag fashion on a sidewalk, or a car weaving in and out of traffic. Detecting trajectory anomalies requires inspecting longer segments of video.

1.2.4 Group Anomalies

Group anomalies can be thought of as unusual object interaction in a scene. An example is a group of persons walking in a formation (such as a marching band). Detecting group anomalies requires analyzing the relationship between two or more regions of video.

1.2.5 Time-of-Day Anomalies

This type of anomaly is orthogonal to all of the other types. What makes these activities anomalous is when they happen. These anomalies are in spirit very similar to the location-dependent anomalies discussed earlier, with the “relevant contextual frame” being temporal instead of spatial. An example is when persons enter a movie theatre at the early hours of dawn. Usually, detecting these anomalies just requires using a different model of normality for different times of day.

A note on the types of anomalies

Not all of these different types of anomalies may be necessary to detect for every application. Thus, video anomaly detection is further, application dependent. In fact, in the publicly available datasets for video anomaly detection that we describe, mainly only appearance-only and short-term motion-only anomalies occur. We should also note that the different types of anomalies are not mutually exclusive. In fact, it can be hard to come up with examples that are exclusive to some of the types listed above.

Anomalousness is a continuous measure

It is important to note that although often discussed in the binary sense, anomalousness is a fluid concept. Every activity is anomalous to some extent. For example, in a scene of a pedestrian walkway, a tall man in a red shirt walking at 1 meter/second may not have been seen exactly in the normal video, but he is most likely similar to some pedestrians in the normal video and therefore should be assigned a low anomaly score. However, if the man is 3 meters tall or walking at 10 meters/second then this should presumably receive a much higher anomaly score. Finding features and distance measures that correspond to our intuitive notions of when two activities are similar is a key to creating successful video anomaly detection algorithms.

1.3 Other Considerations for Video Anomaly Detection

Here we discuss some other characteristics of video anomaly detection formulations that vary in past work on the topic.

1.3.1 Unsupervised, Semi-supervised, Weakly Supervised or Supervised?

The anomaly detection problem is difficult to neatly characterize. Should it be called unsupervised because examples of anomalies have not been provided for supervision? Or should it be called supervised because normal data is provided for supervision? Or how about semi-supervised because only selective data (normal) is provided for training? Some call this problem weakly-supervised because an auxiliary dataset is necessary to determine (provide supervision for) an anomaly score threshold or because proxy labels are often used. We discuss a third possible formulation that some have considered in Section 2.4.5. We assert that summarizing the formulation with these terms is suboptimal and causes confusion for readers. We recommend that future video anomaly detection research works should always provide a full problem formulation to avoid any ambiguities and new methods compare against methods that follow compatible formulations.

1.3.2 Temporal Localization or Spatial Localization Too?

Although several past works have focused solely on the temporal (frame-level) localization aspect of this problem ([10], [17], [9], [8]), we contend that spatial localization is paramount to a useful algorithm. Solely temporal localization is useful for very limited applications such as key-frame prediction and video compression, but even in these cases it is useful to know which parts of the frame were deemed anomalous. In general, in a busy scene, knowing exactly what triggered the anomaly detector. We clearly outline which past works focus solely on temporal localization and which include spatial localization as well.

1.4 Other Surveys

Two past surveys focus on crowded scene analysis [18], [19], which is important and relevant to successful video anomaly detection, but these surveys are not primarily concerned with video anomaly detection. A survey by Sodemann et al. [20] focused on anomaly detection in surveillance videos, but this survey was written before the deep learning explosion in 2012 and thus does not cover important recent work in the area. Two more short surveys focus only on deep learning based methods for video anomaly detection [21], [22], but a large amount of past work does not leverage deep learning and many algorithms that do still depend on integral components that are not deep-learning-based such as [23], [24], [25], [26], [27], [28]. Recently, research in video anomaly detection has picked up pace, and a comprehensive and dedicated survey for this topic is much needed in order to organize and summarize this body of work.

The rest of this article is organized as follows. In Section 2 we describe the publicly available benchmark datasets along with the evaluation protocol for video anomaly detection and set up a taxonomy for the rest of the paper. In Sections 3, 4, and 5 we describe notable past works which have employed different approaches to video anomaly detection. Finally, in Section 6 we discuss the state of research in this field and provide some recommendations for future research directions.
2 OVERVIEW OF SINGLE-VIEW VIDEO OBJECT DETECTION

2.1 Datasets

Benchmark datasets play an important role in the progress of research for any problem in computer vision. They help to define the scope of the problem as well as provide a way to fairly compare the characteristics of different algorithms. For video anomaly detection, there are a handful of publicly available benchmark datasets in common use. We describe them here and provide recommendations based on ground-truth annotation style, size and overall utility of the datasets.

Table 1 provides a summary of the characteristics of these datasets and Figure 2 shows one normal frame and one frame with a single anomaly for each of the datasets we recommend for use.

| Dataset     | Total Frames | Training Frames | Testing Frames | Anomalous Events | Pixel-wise annotation | Track ID annotation |
|-------------|--------------|-----------------|----------------|------------------|----------------------|-------------------|
| Subway      | 22,500       | N/Y             | N/Y            | 9,350            | Y                    | N                 |
| UMN*        | 3,853        | N/A             | N/A            | 85               | N                    | N                 |
| UCSD Ped1, Ped2 | 20,300  | 9,200           | 11,100         | 146,410          | N                    | N                 |
| UCSD Subway | 30,652       | 15,328          | 15,324         | 15,328           | Y                    | Y                 |
| UCSD Street Scene | 47,162  | 24,099          | 23,063         | 56,847           | N                    | Y                 |
| UCSD Subway | 118,400      | 59,200          | 59,200         | 203,257          | N                    | Y                 |

2.1.1 Subway

The Subway dataset [4] is comprised of two long videos of two different indoor scenes, a subway entrance and an exit, making for 2 separate datasets. It mainly captures people entering or leaving through turnstiles. Anomalies include people jumping or squeezing through turnstiles, a janitor cleaning the walls and people walking in the wrong direction. It is unclear at what frame rate one should extract the dataset from these videos and exactly which frames are labeled as anomalous and which frames to use for training and testing. Table 1 uses 15 frames/sec to obtain the frame counts. No spatial ground truth is provided. The datasets contain 85 total anomalous events labeled temporally. These datasets are now quite old and because of the ambiguities and lack of spatial annotation, we do not recommend using these for evaluating an anomaly detection method in any formal capacity. Those seeking the datasets should contact the author directly.

2.1.2 UMN

The UMN dataset [5] has 11 short clips from 3 different cameras at an outdoor field, an outdoor courtyard and an indoor foyer. All clips start with normal activity followed by an anomalous event where the crowd suddenly disperses quickly, hinting at an evacuation scenario. The anomalies are staged and every clip has exactly one anomalous event. There is no clear specification about frame rate for extraction or a training or test split. Fifteen frames/sec was used for the frame counts in Table 1. Additionally, anomalies are only labeled temporally. Because of these ambiguities and the lack of spatial annotation, we do not recommend using it for evaluating an anomaly detection method in any formal capacity.

The dataset and ground truth can be found at http://mha.cs.umn.edu/proj_events.shtml#crowd

2.1.3 UCSD Pedestrian

The most widely used datasets for video anomaly detection are the UCSD Ped1 and Ped2 datasets [29], [2]. Each of these datasets contains videos from a different static camera overlooking a pedestrian walkway, and the crowd density is sometimes high to the point of causing severe occlusions. In this dataset, all non-pedestrian objects as well as unusual motion by pedestrians are deemed anomalous. The types of anomalies present are “biker”, “skater”, “cart”, “wheelchair”, “walk across”, and “other”. UCSD Ped1 consists of 34 training videos and 36 testing videos at a low spatial resolution of 158 × 238 pixels. The field-of-view can be considered mid-range and there are 200 frames per video. UCSD Ped2 contains 16 training and 12 testing videos of slightly higher resolution, 240 × 360 pixels, with 120 to 200 frames per video.

These datasets can be found at http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm. Both spatial (at the pixel-level) and temporal annotation are available for UCSD Ped1 and Ped2 datasets from the authors. One should note that the authors only released partial pixel-wise ground truth for UCSD Ped1, which was subsequently completed by the authors of [30] and made available at https://hci.iwr.uni-heidelberg.de/COMPVIS/research/parsing/. Very recently, the authors in [31] released their “corrected” set of pixel-level annotations as well, claiming that the original annotation has errors at https://github.com/SeaOtter/vad_gan Another set of bounding box annotations containing anomalous region identifiers as well as track identifiers required for evaluating using a more recent criteria has been made available by the authors of [28] at http://www.merl.com/demos/video-anomaly-detection

2.1.4 CUHK Avenue

The CUHK Avenue dataset [4] consists of short video clips taken from a single camera looking at the side of a building with pedestrian walkways by it. The videos mainly contain people walking in and out of a building. Concrete columns that are part of the building cause some severe occlusion. The dataset contains 16 training videos and 21 testing videos of spatial resolution 640 × 360 pixels. There are a total of 47 anomalous events which are mostly staged and comprise actions such as “throwing papers”, “throwing bag”, “child skipping”, “wrong direction” and “bag on grass”. We should note that this was the first dataset to introduce a loitering (static) anomaly with the bag, which is extremely important for surveillance applications.

Both temporal and pixel-level (in bounding box form) annotations are provided by the authors. The dataset and ground truth can be found at http://www.cse.cuhk.edu.hk/leojia/projects/detectabnormal/dataset.html. Another set of bounding box annotations containing anomalous region identifiers as well as track identifiers required for evaluating using more recent protocol has been made available by the authors of [28] at http://www.merl.com/demos/video-anomaly-detection

Researchers should be aware that some papers that report results on Avenue used some evaluation code available on GitHub [https://alliedel.github.io/anomalydetection/] that incorrectly computes pixel-level results [12], [23], [14].
The code produced pixel-level area under the curve (AUC) numbers that were higher than frame-level AUC numbers, which is not possible since frame-level AUC imposes an upper bound on pixel-level AUC. Future papers should not cite these incorrect results and should not use the buggy code that produced them.

2.1.5 Street Scene

The largest dataset, Street Scene [6], is the most recent addition to the publicly available datasets for video anomaly detection. Street Scene consists of 46 training and 35 testing high resolution $1280 \times 720$ video sequences taken from a USB camera overlooking a scene of a two-lane street with bike lanes and pedestrian sidewalks during daytime. The dataset is challenging because of the variety of activity taking place such as cars driving, turning, stopping and parking; pedestrians walking, jogging and pushing strollers; and bikers riding in bike lanes. In addition the videos contain changing shadows, moving background such as a flag and trees blowing in the wind, and occlusions caused by trees and large vehicles. There are a total of 56,847 frames for training and 146,410 frames for testing, extracted from the original videos at 15 frames per second. The dataset contains a total of 205 naturally occurring anomalous events ranging from illegal activities such as jaywalking and illegal U-turns to simply those that do not occur in the training set such as pets being walked and a metermaid ticketing a car. We refer readers to [6] for a more detailed description with complete meta-data.

The authors make the dataset available along with a set of bounding box annotations containing anomalous region identifiers as well as track identifiers required for evaluating on more recent protocols (that they also introduced) at http://www.merl.com/demos/video-anomaly-detection.

2.1.6 Other Datasets

It is worth noting two other datasets that are useful for multi-scene video anomaly detection. Because these datasets include videos from various unconnected scenes and from which a single model is meant to be learned, they are not applicable to the single-scene video anomaly detection formulation that this survey focuses on.

ShanghaiTech

ShanghaiTech [9] is a recent contribution that contains videos from 13 different scenes. A typical video has people walking along a sidewalk of a university. Anomalous activity includes bikers, skateboarders and people fighting. The dataset is intended to be used to learn a single model from the training sets of all 13 scenes. While it is conceivable to treat this dataset as 13 separate datasets (as with UCSD Ped 1 and Ped2), this is problematic since this would yield an average of 10 anomalous events per scene which is very small, and it is not clear whether the variation captured in each scene’s small training set is meant to serve as representative of normal activity.

UCF-Crime

The UCF-Crime dataset [8] is a recently proposed new dataset for video anomaly detection. This dataset contains 128 hours of internet videos taken from many different cameras and contains criminal anomalous activities such as burglary, shoplifting and assault. Anomalies are only annotated temporally (i.e. no spatial annotations are available). The authors also advocate for classifying anomalies according to a predetermined set of anomaly types which makes the problem formulation that this dataset is intended for different from the usual multi-scene video anomaly detection formulation.

2.2 Evaluation Protocol

It is important to remember that anomalies are scene-dependent and what is anomalous is completely determined by what activity occurs at test time but is missing from the training set (that defines normal activity). Moreover, the ground truth annotations are binary in nature although anomalousness is a fluid notion. Determining which activities are missing from the training video can often lead to ambiguities. For example, two people walking next to each other along a sidewalk may exist in the training video, but two people holding hands while walking may not. Should the latter be marked as anomalous? In which frame exactly
does the anomaly begin and end? Should the entire area including both pedestrians be marked as anomalous or just a tight area around the hand-holding? Every dataset and annotator for this task is imperfect and ambiguities such as these will exist. Ideally, an evaluation measure would attempt to give a realistic measure of the qualitative performance of an algorithm in practice in light of inevitable ambiguities in labeling.

2.2.1 Traditional Criteria

Traditionally, research in this field has used frame-level and pixel-level area-under-the-curve (AUC) criteria to evaluate performance, first described in [29], which also presented the UCSD Pedestrian datasets. In [29], the authors fail to fully describe the evaluation measures. Specifically, in [29], the authors define a true positive as a frame where at least 40% of the truly anomalous pixels in the frame are detected, and a false positive otherwise. In their subsequent work [2], they clarify that a false positive can only be counted for frames that do not contain any anomaly annotation, that is, a false positive should not be counted when fewer than 40% of the pixels are detected in a frame that has an anomaly. The clarification makes for a strict reduction in the counts of false positives. We believe that some earlier works might have reported results under the incorrect interpretation of this evaluation metric, leading to much lower pixel AUCs being reported. Here we restate this clarified version.

The frame-level criterion is as follows: Detected frames are designated as those with at least one pixel in a frame that received a score larger than a given anomaly score threshold. A detected frame is counted as a true positive if the frame was annotated as anomalous according to ground truth and a false positive otherwise. The total number of positives and negatives are determined by the frame-level annotations and are used to compute true positive and false positive rates. The frame-level criterion does not evaluate whether sufficient spatial localization has been achieved and the authors themselves recommended against using solely this criterion in [2].

The pixel-level criterion is as follows: Detections are all pixels in a frame which received anomaly scores greater than a given anomaly score threshold. A true positive is counted if over 40% of anomaly-annotated pixels in a frame are detected. If a frame has no anomaly annotation and yet even one pixel is detected, a false positive is counted. Notice that with this criterion, even though spatial localization is taken into account (albeit crudely), the counting of true positives and false positives is still at the frame level. The total number of positives and negatives are as with the frame-level criterion. This has the following consequences:

1) A frame can be counted for only one true positive even if there are multiple anomalies present in the frame. The 40% threshold is applied over all annotated pixels in a frame.

2) A frame that contains an anomaly annotation does not count for a false positive regardless of any incorrect detections present in the frame.

3) A frame without an anomaly annotation can be counted for only one false positive even if there are multiple distinct detected regions in a frame.

4) The criterion does not penalize looseness of a detection. That is, as long as 40% of annotated pixels are detected, it does not hurt performance to change the detection mask to the entire frame.

A threshold on anomaly score is varied in order to generate Receiver Operating Characteristic (ROC) curves of false positive rate versus true positive rate. Area under the ROC curve (AUC) or Equal Error Rate (EER) is used to summarize an ROC curve.

Notice that as described, frame-level AUC for a method imposes an upper-bound on pixel-level AUC. As the authors in [6] observe, points 2 and 3 above admit a simple post-processing step that makes pixel-level AUC exactly reach its upper bound: dilating detection masks with a filter of the same size as the frame (i.e. if a single pixel is detected as anomalous in a frame, make all pixels in the frame anomalous). This can only increase the detection rate without changing the false positive rate according to the pixel-level criterion.

Although these criteria can be useful for ranking different video anomaly detection algorithms, they are now saturated on the smaller datasets (frame-level AUCs have repeatedly been reported on the UMN dataset at > 99% for the past few years) and clearly have serious flaws.

2.2.2 Recent Criteria

Several researchers have recognized these drawbacks of the frame-level and pixel-level criteria and a few have attempted to propose new criteria aimed at addressing them. The authors of [24] proposed the Dual Pixel Level criterion which adds an additional constraint to the pixel-level criterion. In addition to the detected pixels needing to cover at least 40% of the annotated anomalous pixels, at least 10% of the detected pixels need to be covered by the annotated anomalous pixels. In other words, the detected pixels cannot include too many normal pixels (thus preventing the post-processing filtering mentioned above from helping). While this is an improvement, it still cannot correctly count true positives and false positives in frames with (a) multiple anomalies, (b) both true positive as well as false positive detections and (c) multiple false positive detections. The authors of [33] also realized that the pixel-level criterion is flawed and used object-detection style Intersection Over Union (IOU) to penalize both tightness and looseness of a detection on the CUHK Avenue dataset. Unfortunately, this does not fix the issues with multiple counts of either true positives or false positives. Moreover, they are not able to use this IOU-based criterion on other datasets due to differences in annotation formats.

The authors of [6] proposed two new criteria, region-based and track-based, to replace the previous criteria. The new criteria are claimed to provide a much more realistic picture of how an algorithm will perform in practice. They take the perspective that the evaluation protocol should be designed in such a way as to account for ambiguities, biases and inconsistencies that are to be expected in any anomaly detection dataset. To fix the issues with the old criteria, they essentially take two steps:

1) They account for inherent ambiguity in labeling and detecting anomalous events by suggesting a loose
object detection style Intersection Over Union (IOU) criterion to judge spatial localization. Further, their track-based criterion only requires that anomalies in a fixed percentage of frames in an anomalous track be detected.

2) They count true and false positives atomic to a detection rather than atomic to a frame. This means that under their criteria, a frame can have more than one true or false positive, in line with basic intuition.

The region-based criterion is as follows: Detections are all pixels with score greater than a given anomaly score threshold. A detected region is a connected component of detected pixels. A true positive is counted if a detected region has IOU of at least $\beta$ with a ground truth bounding box. A false positive is counted for every detected region that does not satisfy an IOU of $\beta$ with any ground truth bounding box. Further, a detected region may account for more than one true positive count by spanning multiple ground truth bounding boxes, but a ground truth bounding box may only account for one true positive count. To account for spatial annotation ambiguities as well as for the fact that a single detection region might span more than one bounding box annotation, they suggest setting $\beta$ to a low value of 0.1, which seems to work well in practice on existing datasets. The detection rate is computed as the number of true positives divided by the number of ground-truth annotated bounding boxes. The false positive rate per frame is computed as the total number of false positives divided by the number of testing frames.

The track-based criterion is as follows: Detected regions are defined as for the region-based criterion as connected components of pixels with above-threshold anomaly scores. An anomaly-annotated track is detected if $\alpha\%$ of anomaly annotations in the anomalous track overlap with detected regions at a spatial IOU of at least $\beta$. False positives are as with the region-based criterion, i.e. detected regions that do not overlap (IOU $\geq \beta$) with any ground truth bounding boxes. The detection rate for the track-based criterion is the fraction of annotated anomalous tracks that are successfully detected. False positives are counted per frame as with the region-based criterion. The authors suggest setting $\alpha = 10$ initially to account for ambiguities in temporal annotation. This criterion is currently the most reflective of real-world anomaly detection performance. Another nice property is that the $\alpha$ used to report results can be increased as this measure starts to saturate on existing datasets and research shifts focus to tougher datasets.

Notice that since false positives are counted on a per frame basis, the maximum possible false positive rate for either criterion can exceed 1.0. The authors recommend summarizing the ROC curve by calculating AUC for false positive rates per frame from 0 to 1.0 for both criteria.

As a consequence of using these new criteria, bounding box annotations with unique anomaly IDs as well as track IDs are required, which the authors provide for the UCSD Ped1, UCSD Ped2, CUHK Avenue and Street Scene datasets.

Finally, one should also consider that measures such as AUC only provide a summary of a narrow view of performance, and have many drawbacks [34]. For these reasons, researchers are encouraged to provide qualitative analysis and visualizations of detections. Of particular importance is the quality of false positives predicted by different methods, which cannot possibly be captured without visual inspection. A method that produces false positives in test data corresponding to plausibly odd behaviors (that did not exist in the training data) should be favored to another that produces seemingly random false positives, when otherwise numerical measures such as AUCs are comparable between them.

2.3 A Taxonomy of Video Anomaly Detection Approaches

At a high level, past video anomaly detection work can be categorized into distance-based, probabilistic and reconstruction-based approaches. See Figure 3 for intuition on how these approaches work and the subtle similarities and differences between them. Here we review popular works that evaluate performance on at least one of the aforementioned video anomaly detection benchmark datasets, but also give some treatment to seminal works in the area. These approaches are not mutually exclusive, as methods that seem to operate in a distance-based fashion at first sight could easily have probabilistic interpretations; the categorization is merely for convenience. Based off of the basic intuition behind video anomaly detection as explained in Figure 3, we further group methods by both the representation and modeling strategies they employ.

2.3.1 Broad Themes in Representation

Broadly, there are two classes of representations used by video anomaly detection approaches, hand-crafted features and deep features from a CNN. Hand-crafted features include spatio-temporal gradients [3, 32], dynamic textures [29, 2], histogram of gradients [35, 26, 10], histogram of flows [10, 33, 37, 38], flow fields [4, 39, 40, 30, 41], dense trajectories [34] and foreground masks [30, 31]. The deep features are further either extracted as-is from a pre-trained network (such as [23, 26, 42, 43, 27, 32]) or are learned while optimizing for a particular task related to anomaly detection, such as with auto-encoders optimizing for low reconstruction error (such as [23, 44, 24, 10, 17, 45, 11]).

Another consideration in representation is the atomic unit of processing. Algorithms process atomic units ranging from image patches (such as [30, 37, 25]) to video patches (such as [4, 10, 46, 47, 48, 29, 35, 3, 2, 53, 44, 24, 32, 10, 28, 11, 22]) to single full frames (such as [23, 26, 49]) and even video snippets (short sequences of full frames) (such as [39, 35, 43, 27, 10, 17, 2]) when dealing with image or video patches, algorithms operate on units from single fixed-size patches (such as [41, 35, 6]) to multi-scale fixed-size patches (such as [25, 30, 41]) to arbitrarily-sized region proposals (such as [50]).

2.3.2 Broad Themes in Modeling

Broad themes in modeling are use of one-class SVMs (such as [36, 25, 32, 32, 32]), nearest neighbor approaches (such as [11, 38, 26, 6, 28]), Hidden Markov Models (such as [46, 47, 48, 2]) and more generally Probabilistic Graphical Models (such as [30, 41, 2]). More recently,
Fig. 3. An overview of the 3 basic approaches past work has taken to video anomaly detection.

deep learning approaches have started using adversarial training strategies (such as [49], [45], [9]).

Some works focus solely on frame-level (temporal) localization, and in most cases, this means that this objective is built into the model, and as such, the models fail to perform adequate spatial localization ([10], [21], [8]). For instance, methods that use video snippets as their atomic unit of processing often also have a temporal detection focus.

Some works do not specifically account for the location-dependent nature of anomalies, such as [25], [37], [10], [17], [49], [42], [9], [11]. For example, methods that use full frames or video snippets as their atomic unit of processing often overlook this characterization. That is, these methods would not be able to distinguish loitering outside an embassy building from loitering in a public park beside it; they operate under a looser definition of anomalies than that provided in Definition 1. Others account for the location-specific nature of anomalies in one of two ways: (1) scoring voxels conditioned on their location in the camera frame (such as [4], [35], [6], [28]) (2) providing additional context in the form of information from neighboring voxels for scoring (such as [48], [2], [41]).

Another problematic practice that has emerged is that of per-video normalization, such as that in [10], [42], [17], [49], [9]. Here, for every testing sequence, abnormality scores are assigned per frame and subsequently min-max normalized using scores within the same video. This practice has the inherent assumption that every test sequence has at least one normal and one anomalous frame. This is further problematic because scores assigned to frames across videos are not comparable anymore and this does not reflect the way real unseen data would have to be scored - the “end of a test video sequence” is unknown in practice.

2.4.2 Supervised anomaly detection

This approach assumes that anomalous data is available during training time. There are two main problems with this assumption: (1) all possible future anomalous activities cannot possibly be available and annotated in any natural scene, especially given that they occur so rarely (2) even if all possible anomalous activities were available for supervision, the problem itself would reduce to binary video classification where the anomalous class is “known”. This defeats the spirit of video anomaly detection where the ultimate goal in practice is to detect any deviation from normality. For a specific example, consider the work of [53]. The authors build a simple spatio-temporal CNN classifier to perform the normal/anomaly classification on fixed-size
video patches. To achieve this, they use data from both the normal and anomalous classes for training.

### 2.4.3 Video-level weak supervision

This approach relies on having weak supervision in the training of a model in the form of video-level labels (as opposed to snippet/frame-level labels). As far as we know, this approach was mainly born out of the introduction of the UCF-Crime dataset which has video-level labels in both the training and test sets. While it has utility, this problem formulation seems to be an overly specific one. The immediate concern is as with the supervised setting: how can one expect to have videos of all possible anomalous activities at training time when they occur so rarely and are susceptible to concept drift?

The authors of [8] along with the presentation of the UCF-Crime dataset, discuss a Multiple Instance Learning (MIL) Framework for performing anomaly detection using weak supervision of this form. Bags contain fixed-length snippets of videos, where positive bags contain at least one anomalous snippet and negative ones none. They perform MIL ranking by enforcing a constraint that the maximum score over snippets in a positive bag must be greater than a negative bag and add additional sparsity and temporal smoothness constraints to provide better priors to the classification task. They present the first method that utilizes video-level labels for video anomaly detection, so they are able to compare only against methods that cannot utilize these labels. Zhu [54] follows up on this work; they learn a motion-aware feature and demonstrate that it can provide large gains in the MIL framework. Very recently, in [55], the authors convert this weakly-supervised formulation to a fully supervised one with noisy labels, where the primary task becomes to clean the noise and the secondary task of video anomaly detection is converted to binary video action recognition. They use a graph convolutional network [50] to clean the noise in an alternating optimization mechanism.

### 3 Distance-based Approaches

Distance-based approaches involve using the training data to create a model of “normality” and measuring deviations from this model to determine anomaly scores. Usually, these models are themselves quite simple, but clever representations and formulation lead to good performance. See Figure 9 (a), distance-based approaches can be seen as a more general form of both probabilistic and reconstruction-based approaches.

In [35], the authors take the premise that anomalies have local spatio-temporal “signatures”, causing them to have low likelihood under a joint probability distribution of local normal data. They extract overlapping fixed-size video patches and represent them with low-level motion descriptors. They compute statistics using spatio-temporal filters on these representations and calculate K-nearest neighbors (K-NN) distance within each training and test video, for each video patch at every location. They then compute a composite score by aggregating weighted K-NN distances. Composite scores for all video patches from training and testing videos are ranked to perform final detection.

In [36], the authors extract a set of social force [57], HOG (histogram of gradients) [58], HOF (histogram of optical flow) [59], MBH (motion boundary histogram) features [60] and dense trajectories [61] from video snippets. They use Vector Quantization (VQ) coding to represent the features and a one-class SVM [62], with either linear, RBF or polynomial kernels to perform anomaly detection.

In [25], the authors propose one of the first approaches that used learned representations with deep networks for video anomaly detection. They use two streams (RGB and optical flow) of stacked denoising auto-encoders (DAE) on multi-scale fixed-size overlapping video patches to learn low-dimensional representations. They then use the latent codes from the DAEs in a one-class SVM [62] with an RBF kernel to perform one-class classification for anomaly detection. They further present two ways to perform fusion between the modalities, at the representation stage and at the scoring stage.

In [44], the authors train a simple 2-layer sparse autoencoder to reconstruct non-overlapping fixed-size video patches from training video. They consider the sparse representation layer as a “global” descriptor. For “local” descriptors, they represent each patch with a vector of Structural Similarity Index Measures (SSIM) [63] to its spatio-temporal neighbors. They detect anomalies by computing the Mahalanobis distances to training Gaussians estimated (similar to [24], [43]) from both feature representations and only designate anomalies if they have large Mahalanobis distances under both Gaussians.

In [23], the authors first partition video into small fixed-size video patches. In stage 1 of a cascaded scheme, they use a simple 2-layer sparse auto-encoder and embed weak Gaussian classifiers at the latent layers to reject normal patches. They consider the small patches remaining as spatio-temporal interest points to extract larger video patches from their neighborhood for stage 2, which consists of a deeper 4-layer sparse auto-encoder trained in a layer-wise fashion embedded with similar weak Gaussian classifiers in the latent layers. Those video patches left unclassified into the normal class are scored with the Mahalanobis distances each intermediate representation received and considered an anomaly if all 4 Mahalanobis distances are greater than a threshold.

**23** presents one of the first approaches that makes use of features from a pre-trained CNN for video anomaly detection. This is one of the only approaches to use single frames as their atomic processing unit. They train a one-class SVM [62] with a linear kernel on deep features extracted from a VGG-f network on each mean-subtracted frame [64]. They smooth their score maps with a spatio-temporal filter and perform localization by dividing video into fixed-size video patches and simply aggregating anomaly scores over the patch regions.

In [65], the authors present a way to use convolutional winner-take-all auto-encoders [65] to learn motion-feature representations from optical flow fields of fixed-size video patches. They then use the learnt motion-feature representations to build location-dependent one-class SVMs [62] to perform anomaly scoring.

In [67], the authors present a unique geometric approach to anomaly detection. They use dense trajectories
from training frames to create an ensemble of extended convex hulls [68], identifying anomalies at test time using a polytope inclusion test, presumably scoring individual trajectories using their distance-to-convex-hull. They also cluster potentially anomalous trajectories to detect anomalous regions and filter out small false positive detections.

In [69], the authors build a model of normality using the Growing Neural Gas [20] algorithm on STIP features [71] extracted from video snippets/patches. They contend that past methods have not sufficiently dealt with “changing scenes” and propose augmenting the GNG model with online updates in the form of neuron insertion, deletion, learning rate adaptation and stopping criteria. Detection is performed by simply determining whether new patterns are significantly different from nearest-neighbor in the GNG model by studying the distribution of distances.

[43] also use features from a pre-trained deep network, but in a two-step cascaded anomaly detection method. First, they extract feature maps of a video snippet from a pre-trained CNN and evaluate it’s Mahalanobis distance under a Gaussian estimated from feature maps on training data. They produce score maps at this stage by simply rolling-back on the receptive field of pixels in the feature maps. For those Mahalanobis distances in an intermediate range, they further pass location-dependent fixed-size crops of feature maps into a sparse auto-encoder trained on such crops from training data to produce more discriminative features, and use the Mahalanobis distance under a similar Gaussian distribution assumption to further distinguish anomalies.

In [27], the authors present another way to use image features from a pre-trained convolutional network, AlexNet [22]. They also propose a two-stream model, operating on both appearance features and optical flow fields. Using the CNN-extracted features, they apply Iterative Quantization Hashing [23] via a pre-trained binary fully convolutional network to generate binary maps for each frame. They then develop a Temporal CNN Pattern (TCP) measure, a statistical measure of the amount of change of the appearance features over time. Fusion of the two streams produces their final anomaly score maps.

In [45], the authors present one of the first approaches to use adversarial training for video anomaly detection. They use a discriminator network (D) tasked with distinguishing original image patches from reconstructions of noisy patches obtained from a denoising auto-encoder network (R) which plays the role of generator. Since R is trained only on image patches from training data, it decimates outliers and thus enables D to tell an anomalous image patch from its reconstruction easily.

In [32], the authors propose a two-stage anomaly detection algorithm. They extract fixed size video patches from training video, augment them with location, appearance (extracting feature maps from a pre-trained CNN) and motion information (in the form of 3D gradients). For first stage detection, they perform K-means clustering and eliminate small clusters corresponding to noise/outliers to create a robust representation. Second stage detection involves building K one-class SVMs (one for each cluster) to create a “narrowed normality clusters” model, and at test time treating the maximum score for a test patch under these K one-class SVMs as the abnormality score.

In [11], the authors convert the anomaly detection problem to k multi-class 1-versus-rest classification problems, building on their previous work [52]. They use feature pyramid networks [24] to extract crops, train convolutional auto-encoders on appearance and gradient features of these crops to learn latent representations and then perform k-means clustering followed by training of k one class SVMs to make binary one-versus-rest classifications. At test time they simply use the inverse of the maximum of k classification scores as an anomaly score. They do not report spatial localization performance.

In [6], the authors present two baseline algorithms for future comparison on their newly released dataset, Street Scene. They use simple nearest neighbor location-dependent anomaly detection scheme using hand-crafted representations of video patches (flow fields or blurred foreground masks) along with hand-crafted distance measurement (a normalized L1 or L2 voxel-wise distance respectively). They vastly reduce the number of distance computations by building a concise representative exemplar model from training data. Interestingly, they show that these simple methods are able to outperform some of the previous state of the art methods on other datasets, possibly indicating that algorithms have developed biases specific to certain datasets.

In [28], the authors build on the simple nearest neighbor scheme by replacing the hand-crafted representation and distance function with learned ones by training a Siamese neural network [25]. The Siamese network is trained to classify video patch pairs as similar or different and is used to find testing video patches that are different from all training video patches and are therefore anomalous. An exemplar model (consisting of all unique normal video patches) is learned from training data of the target dataset. Finally, nearest neighbor scoring between test video patches and exemplars using the trained Siamese network is used to assign anomaly scores to each testing video patch.

4 Probabilistic Approaches

Probabilistic approaches compute distance under a model in some probability space. These methods usually aim to admit modeling into a probabilistic framework such as with probabilistic graphical models (PGMs) or high-dimensional mixtures of probability distributions. See Figure 3(b) for this intuition.

In [4], the authors use fixed-location monitors on the camera frame which have a fixed-size storage buffer in which they store optical flow fields. They declare anomalies as those test optical flow observations with low likelihood given the corresponding monitor’s buffer, which they model either as a histogram of observations or using kernel density estimation.

In [10], the authors introduce the first modeling approach that utilizes the social force model [57]. They place a grid of particles over the video and advect them with optical flow from the video to estimate social force interactions which are roughly the difference between a pixel’s optical flow and the average optical flow in a neighborhood around the pixel. The idea being that the reason a pixel differs from its neighbors is due to interactions among particles. These
social force interactions are mapped onto the image plane to yield a mapping called force flow. They then create a Latent Dirichlet Allocation (LDA) [75] bag-of-words model (where the words are spatio-temporal regions of force flow) on random training video patches in each video snippet and detect anomalies as low-likelihood frames under the model. Localizing is done by simply detecting regions with high force flow.

In [45], the authors compute binary motion labels for each pixel by simple background subtraction. They use spatio-temporal neighborhoods around each pixel to compute co-occurrence statistics on the motion label representation of normal data and use the co-occurrence matrix as the potential function in a Markov Random Field to perform anomaly detection via likelihood ratio testing.

In [47], the authors represent video with spatio-temporal gradients. They use multivariate Gaussians to model their distribution for video patches and a mixture of Gaussians to represent the distribution of video patches for a given location in the camera frame. Finally, they use a coupled Hidden Markov Model to incorporate the effect of spatial and temporal correlations between the video patches.

[48] presents a way to use a spatio-temporal Markov Random Field to model relationships between neighboring training video patches extracted from a grid on video. They represent each video patch as a node in the graph by building a Mixture of Probabilistic Principal Components Analyzers (MoPPCA) [77] on optical flow observations. They detect anomalies by computing a maximum a posteriori estimate of normality at test time. They also show how their model can be incrementally updated to account for environmental changes and concept drift.

In [50], the authors also advect particles on a grid of optical flow observations on video similar to [10], but they focus on trajectories of these particles. They cluster these trajectories and model chaotic dynamics of them using two chaotic invariants. Anomaly detection is performed by simply estimating parameters of a Gaussian mixture model on this chaotic feature set from normal data and evaluating the likelihood of test data.

In [29], the authors propose learning a Mixture of Dynamic Textures (MDT) [78, 29] from training video patches, with the mixtures shared across larger “cell” regions. They detect anomalies as those regions with high center-surround saliency as given by a discriminant saliency criterion [59]. In [2], they build off the MDT representation to operate at multiple scales. They integrate spatial and temporal anomaly scores from multiple scales using a conditional random field [81] framework.

The authors in [30] use a rather unique premise - that anomaly detection must be done indirectly by trying to “explain away” the normality in the test data using information learned from the training data. They seek a video parsing approach that simultaneously discovers foreground object hypotheses that jointly explain the foreground in a frame and those that have matching normal exemplar hypotheses. Those object hypotheses at test time which are necessary to explain the foreground but do not match any exemplar hypotheses from normal training data are anomalous. In [31], they further build on this idea by considering object hypotheses in the form of flexible video pipes instead of just image patches.

In [38], the authors propose a hierarchical local plus global method to detect anomalies. They model video with Spatio-Temporal Interest Point (STIP) features [71] and form a codebook with K-means clustering, detecting local anomalies as those with high distance to the $k^{th}$ nearest neighbor. For global anomalies, they consider ensembles of STIP features to construct a high-level codebook of interaction templates and build Gaussian Process Regression (GPR) models [82] with an RBF kernel for each model. They then designate low-likelihood test ensembles under the $k^{th}$ nearest neighboring GPR ensemble model as anomalous.

In [26], the authors propose a unique method to recount anomalous events as they are detected. They first train a Fast-RCNN [83] model to predict object, action and attribute classes from large-scale COCO [84] and Visual Genome [85] image datasets. Then for each frame, they extract features of each region of interest (RoI) from the second-to-last fully connected layer and perform anomaly detection with either nearest neighbor distance to training sample, a one-class SVM with RBF kernel or likelihood under a kernel density estimate with RBF kernel. Recounting is performed by simply looking at maximal predictions of object, action and attribute classes.

In [60], the authors use PCANet [87] to extract deep representations, learned from 3D gradients of normal image patches. They then use Deep GMMs [88] to model a generative process of normal patterns, maximizing a lower-bound on log-likelihood. The deep GMM model simply yields likelihood scores for testing patterns which are used as anomaly scores.

5 Reconstruction-based Approaches

Reconstruction approaches aim to break down inputs into their common constituent pieces and put them back together to reconstruct the inputs. They are based on the premise that out-of-distribution inputs such as anomalies are inherently harder to reconstruct when compared to in-distribution normal data, thus justifying the use of reconstruction error as a proxy for anomaly score. See Figure 3(c) for an illustration of this intuition.

In [10], the authors train a convolutional auto-encoder to reconstruct training video snippets with a pixel-wise L2 loss. Reconstruction error on testing video snippets, normalized per-video sequence, serves as their abnormality scores. They do not perform spatial localization, claiming a focus on temporal localization. Interestingly, they also train a generalized auto-encoder on training data from several datasets and show that it performs about as well as one trained on a single dataset.

In [17], the authors build on the convolutional auto-encoder architecture in [10] by preserving temporal ordering of frames through the convolutions and modeling the temporal information at the bottleneck layer with specialized convolutional LSTM [89] layers.

In [49], the authors attempt the first use of Generative Adversarial Networks (GANs) [90] for video anomaly detection. They train two conditional GANs, that take as input $(x, z)$ pairs of frames and noise vectors and generate corresponding frames $y$ of a different modality (they use
raw frames to optical flows and vice versa in the two GANs). The discriminators are asked to classify pairs of \((x, y)\) representations of frames as real or fake. Assuming that anomalies are not reconstructed well, they fuse reconstruction errors from both modalities, weighting the optical flow errors twice that of raw frames, and performing per-video normalization to perform anomaly scoring for detection and pixel-wise localization.

In [91], the authors perform feature learning and reconstruction on fixed-size raw video patches using Restricted Boltzmann Machines (RBMs) [92] using the Contrastive Divergence [93] training algorithm. They combine reconstruction errors at test time from different pyramid levels and overlapping patches to come up with an anomaly score.

In [9], the authors contend that predicting a video snippet’s future frame must be harder for anomalous activities compared to normal ones, and thus design a future frame prediction framework. They train a U-net-style network [94] that takes training video snippets of length \(t\) as input and predicts a future frame for time \(t+1\). Further, they use FlowNet [95] to estimate pairs of optical flow maps between the frame at \(t\) and real or reconstructed frames at \(t+1\). L1 losses between flow maps, intensity and directional gradients of reconstructions along with an adversarial loss to differentiate the real and reconstructed frames at \(t+1\), followed by per-video normalization of errors, forms their anomaly score. They also do not report spatial localization performance.

In [96], the authors address the problem by learning a correspondence between common object appearances and their associated motions in a two-stream model. Using a single frame as input, they use a single encoder coupled with both a U-net decoder that predicts motion as well as a devonconcolutional decoder that reconstructs the input frame, governed by \(l_p\) reconstruction error loss terms. They consider this entire network a generator in a conditional GAN, where the discriminator is another small network that distinguishes between pairs of input frames and corresponding real/estimated flow fields which is governed by a binary classification loss. They optimize this cGAN framework in an alternating fashion. For testing frames, they calculate \(l_p\) scores at a patch-level and use per-video normalization of scores for their final frame-level anomaly scores. They also do not report spatial localization performance.

In [31], the authors contend that past reconstruction-based methods have largely operated on low-level features. They seek to address this by performing anomaly detection only with abstract features. First, they train Denoising Auto-encoders (DAEs) on raw video snippets and corresponding flow field representations. They then extract representations at multiple layers and train conditional GANs for each similar to [49]. Lastly, they combine reconstruction error maps from the multiple levels to arrive at a consensus score map for each frame.

In [97], the authors contend that prediction and reconstruction can be combined to exploit advantages and balance disadvantages of both. They seek to do this by creating a generator that operates on video snippets comprised of two consecutive U-net [94] architectures, where the first predicts an intermediate “frame” that is then used by the second to predict the immediate future frame, trained end-to-end by minimizing reconstruction error on intensity and gradient modalities. They also employ an adversarial loss on either ground truth future and predicted future frame pairs or at a finer level similar to PatchGAN [98].

### 5.1 Sparse Reconstruction Approaches

A subset of reconstruction approaches, sparse reconstruction approaches impose an additional constraint on the reconstruction in that it must be performed using a sparse feature set only.

In [37], the authors estimate optical flow fields in video and extract a multi-scale histogram of optical flow features. They then learn a dictionary from these features from training video and use a sparse reconstruction cost based on L1 minimization as their anomaly score on patches from test video. Favorable properties include online update of the dictionary and the ability to define a basis over different representations such as image patches, video patches or video snippets to perform anomaly detection at various levels.

In [3], the authors operate on 3D gradient features of fixed-size video patches extracted from video at multiple scales. They propose sparse combination learning from training video, where the goal is to learn a dictionary of atomic units from training video patches and sets of sparse combinations of these to reconstruct video patches. During test time, the sparse combination with the least reconstruction error is used to score test video patches.

In [42], the authors propose sparse coding with the constraint that temporally close frames be encoded with similar sparse coefficients (TSC). They use a special type of sRNN to enforce this and by optimizing all parameters of this network simultaneously, avoid the non-trivial hyper-parameter search involved in TSC. Interestingly, the representations they operate on are multi-scale pooled features extracted from a pre-trained network on UCF-101 for each full frame. Because they have a focus on temporal anomaly detection, their method does not perform localization.

In [99], the authors propose augmenting a 3D convolutional auto-encoder with a memory module. They argue that this would help overcome some other auto-encoder approaches generalizing “too well” on test data leading to missed detections. At the bottleneck layer, they implement a memory module to using a fixed-size memory with attention-based addressing and hard shrinkage to encourage sparse reconstructions of input video snippets. They also perform per-video normalization and do not report spatial localization performance.

### 6 Table of Approaches

Table 2 lists all of the papers discussed in the previous sections grouped by the type of approach taken, ordered chronologically per approach. The table also summarizes the common types of representation used and the common characteristics of the model of normal activity used. The following list explains the abbreviations used in the table.

**Thematic grouping Table 2 notation guide:**

Proc. unit: Atomic unit of processing.

VS: Video snippets.
| Method | Approach | Proc. unit | Representation theme | Input feats. | Pre-trained net | Model component | Per-video norm. | Modeling theme | Location-dependent | Sp-local. |
|--------|----------|------------|----------------------|--------------|----------------|-----------------|----------------|---------------|------------------|-----------|
| Dist.  | Fixed-size VP | HOG, 3D grad., motion magnitude | | | | OC-SVM | | | |
| Dist.  | VS | HOF, SF, dense trajectories | | | | OC-SVM, AE | | | |
| Dist.  | Fixed-size IP | Raw, flow, deep | | | | OC-SVM, AE | | | |
| Dist.  | Fixed-size VP | Raw, 3MM, deep | | | | AE | | | |
| Dist.  | Fixed-size VP | Raw, deep | | | | AE | | | |
| Dist.  | FF | Raw, deep | | | | OC-SVM | | | |
| Dist.  | Fixed-size VP | flow, deep | | | | OC-SVM, AE | | | |
| Dist.  | FF | dense trajectories | | | | | | | |
| Dist.  | VS, VP | STIP | | | | NN | | | |
| Dist.  | VS | Raw, deep | | | | AE | | | |
| Dist.  | FE, VS | flow, deep | | | | | | | |
| Dist.  | Fixed-size IP | Raw, deep | | | | Adversarial, AE | | | |
| Dist.  | Fixed-size VP | 3D grad., deep | | | | OC-SVM | | | |
| Dist.  | Fixed-size VP | 2D grad., deep | | | | OC-SVM, AE | | | |
| Dist.  | Fixed-size VP | flow, fg-mask | | | | NN | | | |
| Dist.  | Fixed-size VP | flow, deep | | | | NN | | | |
| Prob.  | Fixed-size VP | flow | | | | | | | |
| Prob.  | Fixed-size VP | Flow, social force | | | | | | | |
| Prob.  | Fixed-size VP | Fg-mask, co-occurrence matrix | | | | HMM | | | |
| Prob.  | Fixed-size VP | 3D grad. | | | | HMM | | | |
| Prob.  | Fixed-size VP | flow | | | | HMM | | | |
| Prob.  | VS | flow | | | | | | | |
| Prob.  | Fixed-size VP | DT | | | | | | | |
| Prob.  | Fixed-size IP | Fg-mask, flow | | | | OC-SVM | | | |
| Prob.  | Fixed-size VP | DT | | | | HMM | | | |
| Prob.  | Fixed-size VP | Fg-mask, flow | | | | OC-SVM | | | |
| Prob.  | Fixed-size VP | STIP, 3DSIFT, HOG, HOF | | | | NN | | | |
| Prob.  | Fixed-size VP | 3D grad., HOF | | | | OC-SVM | | | |
| Prob.  | FF | Raw, deep | | | | NN, OC-SVM | | | |
| Prob.  | Fixed-size IP | 3D grad., deep | | | | | | | |
| Recon. | VS | Raw, deep | | | | | | | |
| Recon. | VS | Raw, deep | | | | | | | |
| Recon. | FF | Raw, flow, deep | | | | | | | |
| Recon. | Fixed-size VP | Raw, deep | | | | | | | |
| Recon. | VS | Raw, flow, deep, 2D grad. | | | | | | | |
| Recon. | FE | Raw, flow, deep | | | | | | | |
| Recon. | VS | Raw, flow, deep | | | | | | | |
| Recon. | VS, FE, VS | HOF, flow | | | | | | | |
| Recon. | FF | Flow, deep | | | | | | | |
| Recon. | Fixed-size VP | Deep | | | | | | | |
| Recon. | VS | Raw, deep | | | | | | | |

**FF:** Full frames.
**VP:** Video patch.
**IP:** Image patch.
**VT:** (Flexible) video tube.
**Input feats:** Input feature representation.
**grad.:** gradients, 2D or 3D.
**flow:** Sparse or dense optical flow representation of the processing unit, without binning into histograms.
**deep:** deep features in some form, such as extracted from a pre-trained CNN or learned end-to-end.
**HOG:** Histogram of Oriented Gradients [58].
**HOF:** Histogram of Optical Flow [59].
**MBH:** Motion Boundary Histogram [60].
**Dense trajectories:** [61].
**Social Force:** [57].
**DT:** Mixtures of Dynamic Textures [78], [79].
**STIP:** Spatio-temporal Interest Point features [71].
**3DSIFT:** 3-dimsional Scale Invariant Feature Transform features [101].
**OC-SVM:** Use of one-class SVM [62].
**NN:** Use of nearest neighbor logic.
**HMM:** Use of a vanilla Hidden Markov Model or its more specialized variants such as Markov Random Fields or Conditional Random Fields.
**Adversarial:** Use of an adversarial training procedure in some form.

**AE:** Use of a vanilla auto-encoder or its more specialized variants such as variational, denoising, contractive, or sparse auto-encoders.

**Per-video norm.:** The method performs normalization of anomaly scores per test sequence, encoding an assumption that every test sequence contains at least one normal and one anomalous frame.

**Location-dependent:** Operates in a location-dependent fashion, a local spatial context is taken into account when detecting anomalies. See Section 4.

**Sp-local.:** The method is apparently able to perform spatial localization.

### 7 Discussion

We have provided a comprehensive review of research in single-view video anomaly detection. We built an intuitive taxonomy and situated past research works in relation to each other. We also hope this article will serve to clear up some misconceptions among different problem formulations, use of datasets, evaluation protocol and how to compare against methods that use compatible problem formulation and evaluation schema in their assumptions. For future reference, we provide a compilation of the performance of popular works in the field on the various datasets and
evaluation criteria discussed in Tables 3, 4, and 5. We now provide some best practices and state some observations on the evolution of the field in terms of overarching trends in representation and modeling as they relate to the increasing size of datasets and increasing compute power of devices.

7.1 Best Practices Going Forward
In terms of future best practices, we urge researchers in this area to use the recommended reliable datasets, new evaluation protocol and participate in reproducible research. A qualitative evaluation of quality of false positives is also important. Evaluating on multiple datasets is essential; for example, some works that evaluate solely on UCSD Ped1, UCSD Ped2 and UMN datasets are known to be inherently biased towards the anomalies in these datasets, which are mainly comprised of objects with larger magnitude optical flows. CUHK Avenue and Street Scene have emerged as good supplements with more variation in anomalous activity.

7.2 Trends in Representation
Representation of input to video anomaly detection algorithms was mostly dominated by raw, fixed-size image patches. Some anomalies require analyzing temporal information, so researchers turned to using video patches, which required more compute power. More recently, researchers have started using multi-modal representations of video patches, with raw frames as well as estimated optical flow fields to the point where it is the norm now. Some methods have even attempted to use entire frames and video snippets as input by exploiting advances in GPU compute power. We expect this trend of the increasing complexity of input representation to reverse with the use of 3D and inflated 3D convolutions on raw video (foregoing expensive optical flow field computation) which have become popular in video action recognition [103].

7.3 Trends in Modeling
Meanwhile, modeling has followed a different trend. At first, researchers used very simple hand-crafted features whose distribution could be well modeled with simple assumptions. Soon researchers achieved better results with more complex models, more intricate assumptions and a lot of clever engineering. More recently, the trend has reversed, with a larger reliance on learning representations from data to more directly optimize a cleverly set up optimization scheme and elegant modeling approach. We expect this trend of having the data dominate to continue, especially as larger, more complex datasets become available.

7.4 Looking Ahead
On one hand, video anomaly detection research has come a long way. On the other hand, past research has also neglected tackling some of the more challenging problems in video anomaly detection. In existing datasets, loitering anomalies have not exactly been addressed in specific by modeling. In fact, most past approaches are unable to detect these kinds of anomalies since they rely heavily on motion cues to ignore processing parts of the video. Working on an algorithm to retain the benefits of any recent state of the art method that is also able to detect loitering anomalies is one ripe area for contribution. Another challenge for video anomaly detection methods is the ability to handle rare but normal activity. Such activity, which may appear very sparsely in the normal training video, often causes false positive anomaly detections. An example of such activity is a pedestrian stopping to tie her shoe. This probably does not happen very often and a security guard may not want the anomaly detector to raise an alarm when it does. So the model that is learned from normal video should include not only the most common normal activities but rare, normal activities as well.

In terms of the types of anomalies, group, trajectory and time of day anomalies have largely been unaddressed because benchmark datasets that contain these simply do not exist yet. We urge and expect other researchers to contribute datasets with these properties in the near future.

As researchers move on from a focus on smaller, less complex datasets for which accuracy is becoming saturated, to larger, more complex datasets with a greater variety of anomaly types they will be pushed to invent new video representations and new models that can achieve high detection rates at low false positive rates to make algorithms that are practical for real applications.

BIBLIOGRAPHY

[1] V. Saligrama, J. Konrad, and P.-m. Jodoin, “Video Anomaly Identification,” IEEE Signal Processing Magazine, vol. 27, no. 5, pp. 18–33, Sep. 2010.
[2] Weixin Li, V. Mahadevan, and N. Vasconcelos, “Anomaly Detection and Localization in Crowded Scenes,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 1, pp. 18–32, Jan. 2014.
[3] C. Lu, J. Shi, and J. Jia, “Abnormal Event Detection at 150 FPS in MATLAB,” in IEEE International Conference on Computer Vision (ICCV). Sydney, Australia: IEEE, Dec. 2013, pp. 2720–2727.
[4] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, “Robust Real-Time Unusual Event Detection using Multiple Fixed-Location Monitors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 555–560, Mar. 2008.
[5] [14] Unusual crowd activity dataset of University of Minnesota, available from http://mnh.cs.umn.edu/proj_events.shtml.
[6] B. Ramachandra and M. Jones, “Street Scene: A new dataset and evaluation protocol for video anomaly detection,” in Winter Conference on Applications of Computer Vision (WACV), 2020.
[7] R. Morais, V. Le, T. Tran, B. Saha, M. Mansour, and S. Venkatesh, “Learning regularity in skeleton trajectories for anomaly detection in videos,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 11996–12004.
[8] W. Sultani, C. Chen, and M. Shah, “Real-World Anomaly Detection in Surveillance Videos,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT: IEEE, Jun. 2018, pp. 6479–6488.
[9] W. Liu, W. Luo, D. Lian, and S. Gao, “Future frame prediction for anomaly detection—a new baseline,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 6536–6545.
[10] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, “Learning Temporal Regularity in Video Sequences,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 733–742.
### TABLE 3
Traditional Frame-level and Pixel-level Evaluation Criteria on the UCSD Ped1, UCSD Ped2 and CUHK Avenue Benchmark Datasets from Related Literature, Ordered Chronologically, compiled from this same list. *Some of the earlier works unfortunately use only a partially annotated subset available at the time to report performance.

| Method | UCSD Ped1 | UCSD Ped2 | CUHK Avenue |
|--------|-----------|-----------|-------------|
|        | frame AUC/EER | pixel AUC* | frame AUC/EER | pixel AUC | frame AUC/EER | pixel AUC |
| Adam   | [4] 65.0%/38.0% | 46.1% | 63.0%/42.0% | 18.0% | -/ - |
| Social force | [40] 67.5%/31.0% | 19.7% | 63.0%/42.0% | 21.0% | -/ - |
| MPPCA  | [29] 59.0%/40.0% | 20.5% | 77.0%/30.0% | 14.0% | -/ - |
| Social force + MPPCA | [29] 67.0%/32.0% | 21.3% | 71.0%/36.0% | 21.0% | -/ - |
| MDT    | [29] 81.8%/25.0% | 44.1% | 85.0%/25.0% | 44.0% | -/ - |
| Video parsing | [30] 91.0%/18.0% | 83.6% | 92.0%/14.0% | 76.0% | -/ - |
| Local statistical aggregates | [35] 92.7% | 16.0% | 63.8% | -/ - |
| Detection at 150 FPS | [3] 91.8%/75.0% | -/ - |
| Sparse reconstruction | [32] 86.0%/19.0% | 45.3% | -/ - |
| HMDT CRF | [2] -/17.8% | 82.7% | -/18.5% | -/ - |
| AMDN   | [25] 92.1%/16.0% | 67.2% | 90.8%/17.0% | -/ - |
| ST video parsing | [41] 93.9%/12.9% | 84.2% | 94.6%/10.6% | 81.1% | -/ - |
| App+motion cues | [100] 85.0%/- | 65.0% | 90.0%/- | -/ - |
| Conv-AE | [10] 81.0%/27.9% | - | 90.0%/21.7% | 70.2%/25.1% | -/ - |
| Deep event models | [86] 92.5%/15.1% | - | 96.2%/19.0% | 75.7% | -/ - |
| Compact feature sets | [102] 82.0%/21.1% | 57.0% | 84.0%/19.2% | -/ - |
| Conv-WTA-AE | [65] 91.9%/15.9% | 68.7% | 92.8%/11.2% | 80.9% | -/ - |
| Convolt Kalman filters | [31] 82.3%/23.5% | 66.6% | 99.2%/2.5% | 97.2% | 71.5%/36.4% |

### TABLE 4
Track and Region-based Area Under the ROC Curve for False Positive Rates up to 1.0 on UCSD Ped1, UCSD Ped2 and CUHK Avenue.

| Method | Ped1 track AUC | Ped1 region AUC | Ped2 track AUC | Ped2 region AUC | CUHK Avenue track AUC | CUHK Avenue region AUC |
|--------|----------------|----------------|----------------|----------------|------------------------|------------------------|
| FG masks | 84.6% | 93.0% | 86.0% | 95.0% | -/ - | -/ - |
| Flow | 86.5% | 89.1% | 81.5% | 85.0% | -/ - | -/ - |
| Siamese net | 90.0% | 89.3% | 78.0% | 84.0% | -/ - | -/ - |

### TABLE 5
Track-based, Region-based, Pixel-level, and Frame-level Area Under the ROC Curve on Street Scene.

| Method | Ped1 track AUC | Ped1 region AUC | Ped1 pixel AUC | Ped1 frame AUC | Ped2 track AUC | Ped2 region AUC | Ped2 pixel AUC | Ped2 frame AUC | CUHK Avenue track AUC |
|--------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Autoencoder | 2% | 0.3% | 0.1% | 61% | -/ - | -/ - | -/ - | -/ - | -/ - |
| Dictionary method | 0% | 0% | 0% | 1% | -/ - | -/ - | -/ - | -/ - | -/ - |
| Flow | 52% | 11% | 7% | 48% | -/ - | -/ - | -/ - | -/ - | -/ - |
| FG masks | 53% | 21% | 30% | 61% | -/ - | -/ - | -/ - | -/ - | -/ - |

[12] A. Del Giorno, J. A. Bagnell, and M. Hebert, “A Discriminative Framework for Anomaly Detection in Large Videos,” in *European Conference on Computer Vision (ECCV)*. Springer International Publishing, 2016, vol. 9909, pp. 334–349.

[13] B. Zhao, L. Fei-Fei, and E. P. Xing, “Online detection of unusual events in videos via dynamic sparse coding,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Colorado Springs, CO, USA: IEEE, Jun. 2011, pp. 3313–3320.

[14] R. T. Ionescu, S. Smeureanu, B. Alexe, and M. Popescu, “Unmasking the Abnormal Events in Video,” in *IEEE International Conference on Computer Vision (ICCV)*. Venice: IEEE, Oct. 2017, pp. 2914–2923.

[15] Y. Liu, C.-L. Li, and B. Póczos, “Classifier two sample test for video anomaly detections.” in *British Machine Vision Conference (BMVC)*, 2018, p. 71.

[16] E. Keogh, J. Lin, and A. Fu, “Hot sax: Efficiently finding the most unusual time series subsequence,” in *Proc. of the 5th IEEE International Conference on Data Mining (ICDM)*, 2005, pp. 226–233.

[17] Y. S. Chong and Y. H. Tay, “Abnormal Event Detection in Videos using Spatiotemporal Autoencoder,” *Advances in Neural Networks (ISNN)*, Jan. 2017.

[18] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, “Crowd analysis: a survey,” *Machine Vision and Applications*, vol. 25, no. 3, pp. 367–386, Mar. 2015.

[19] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, “Crowded Scene Analysis: A Survey,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 367–386, Mar. 2015.

[20] A. Sodemann, M. Ross, and B. Borghetti, “A review of anomaly detection in automated surveillance,” *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 42, no. 6, pp. 1257–1272, Nov. 2012.

[21] E. Keogh, J. Lin, and A. Fu, “Hot sax: Efficiently finding the most unusual time series subsequence,” in *Proc. of the 5th IEEE International Conference on Data Mining (ICDM)*, 2005, pp. 226–233.

[22] Y. S. Chong and Y. H. Tay, “Abnormal Event Detection in Videos using Spatiotemporal Autoencoder,” *Advances in Neural Networks (ISNN)*, Jan. 2017.

[23] S. Smeureanu, R. T. Ionescu, M. Popescu, and B. Alexe,
“Deep Appearance Features for Abnormal Behavior Detection in Video,” in *Image Analysis and Processing - ICIAP 2017*. Springer International Publishing, 2017, vol. 10485, pp. 779–789.

M. Sabokrou, M. Fayyaz, M. Fathy, and R. Klette, “Deep-Cascade: Cascading 3D Deep Neural Networks for Fast Anomaly Detection and Localization in Crowded Scenes,” *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1992–2004, Apr. 2017.

D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe, “Learning deep representations of appearance and motion for anomalous event detection,” *British Machine Vision Conference (BMVC)*, 2015.

R. Hinami, T. Mei, and S. Satoh, “Joint Detection and Recounting of Abnormal Events by Learning Deep Generic Knowledge,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2017, pp. 3639–3647.

M. Ravanbakhsh, M. Nabi, H. Mousavi, E. Sangineto, and N. Sebe, “Plug-and-Play CNN for Crowd Motion Analysis: An Application in Abnormal Event Detection,” in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV, Mar. 2018, pp. 1689–1698.

B. Ramachandra, M. Jones, and R. Vatsavai, “Learning a distance function with a siamese network to localize anomalies in videos,” in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020.

V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, “Anomaly detection in crowded scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2010, pp. 1975–1981.

B. Antic and B. Ommer, “Video parsing for abnormality detection,” in *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 2415–2422.

H. Vu, T. D. Nguyen, T. Le, W. Luo, and D. Phung, “Robust Anomaly Detection in Videos Using Multilevel Representations,” *AAAI Conference on Artificial Intelligence*, vol. 33, pp. 5216–5223, Jul. 2019.

R. T. Ionescu, S. Smeureanu, M. Popescu, and B. Alexe, “Detecting Abnormal Events in Video Using Narrowed Normality Clusters,” in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Jan. 2019, pp. 1951–1960.

J. M. Lobo, A. Jiménez-Valverde, and R. Real, “AUC: a misleading measure of the performance of predictive distribution models,” *Global Ecology and Biogeography*, vol. 17, no. 2, pp. 145–151, 2008.

V. Saligrama and Z. Chen, “Video anomaly detection based on local statistical aggregates,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 2112–2119.

K. Ma, M. Doescher, and C. Bodden, “Anomaly Detection In Crowded Scenes Using Dense Trajectories,” 2015.

Y. Cong, J. Yuan, and J. Liu, “Abnormal event detection in crowded sparse scene detection,” *Pattern Recognition*, vol. 46, no. 7, pp. 1851–1864, Jul. 2013.

K.-W. Cheng, Y.-T. Chen, and W.-H. Fang, “Video anomaly detection and localization using hierarchical feature representation and gaussian process regression,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2909–2917.

S. Wu, B. E. Moore, and M. Shah, “Chaotic invariants of Lagrangian particle trajectories for anomaly detection in crowded scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 2054–2060.

R. Mehran, A. Oyama, and M. Shah, “Abnormal crowd behavior detection using social force model,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 935–942.

B. Antic and B. Ommer, “Spatio-temporal Video Parsing for Abnormality Detection,” *arXiv preprint arXiv:1502.06235*, 2015.

H. Luo, W. Shi, and S. Gao, “A Revisit of Sparse Coding Based Anomaly Detection in Stacked RNN Framework,” in *IEEE International Conference on Computer Vision (ICCV)*, Venice, Oct. 2017, pp. 341–349.

M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, “Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes,” *Computer Vision and Image Understanding*, vol. 172, pp. 88–97, Jul. 2018.

M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, “Real-time anomaly detection and localization in crowded scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2015, pp. 56–62.

M. Sabokrou, M. Kholoeei, M. Fathy, and E. Adeli, “Adversarially Learned One-Class Classifier for Novelty Detection,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 3379–3388.

Y. Benezech, P.-M. Jodoin, V. Saligrama, and C. Rosenberger, “Visual events detection based on spatio-temporal co-occurrences,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL: IEEE, Jun. 2009, pp. 2458–2465.

L. Kratz and K. Nishino, “Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, Jun. 2009, pp. 1446–1453.

A. Ahm and K. Grauman, “Learning to locate globally: A space-time MRF for detecting abnormal activities with incremental updates,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, Jun. 2009, pp. 2921–2928.

M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni, and N. Sebe, “Abnormal event detection in videos using generative adversarial nets,” in *2017 IEEE International Conference on Image Processing (ICIP)*, Sep. 2017, pp. 1577–1581.

B. Ramachandra, “Anomaly detection in videos,” *North Carolina State Dept. of Computer Science Ph.D. thesis*, pp. 52–59, 2019.

H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “Rmpe: Regional multi-person pose estimation,” in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2334–2343.

K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

S. Zhou, W. Shen, D. Feng, Y. Wei, and Z. Zhang, “Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes,” *Signal Processing: Image Communication*, vol. 47, pp. 358–368, Sep. 2016.

Y. Zhu, “Motion-Aware Feature for Improved Video Anomaly Detection,” in *British Machine Vision Conference (BMVC)*, 2019, p. 12.

J.-X. Zhong, N. Li, S. Liu, T. H. Li, and G. Li, “Graph Convolutional Label Noise Cleaner: Train a Plug-And-Play Action Classifier for Anomaly Detection,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

T. N. Krip and M. Welging, “Unsupervised classification with graph convolutional networks,” in *International Conference on Learning Representations*, 2017.

D. Helbing and P. Molnar, “Social force model for pedestrian dynamics,” *Physical Review E*, vol. 51, no. 5, p. 4282, 1995.

N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

N. Dalal, B. Triggs, and C. Schmid, “Human detection using oriented histograms of flow and appearance,” in *European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 428–441.

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, “Dense trajectories and motion boundary descriptors for action recognition,” *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, “Support vector method for novelty detection,” in *Advances in Neural Information Processing Systems*, 2000, pp. 582–588.

Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli et al., “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” *arXiv preprint arXiv:1405.3531*, 2014.

H. Tran and D. Hogg, “Anomaly Detection using a Convolutional Winner-Take-All Autoencoders,” in *British Machine Vision Conference (BMVC)*. London, UK: British Machine Vision Association, 2017, p. 139.

A. Makhzani and B. J. Frey, “Winner-take-all autoencoders,” in *Advances in Neural Information Processing Systems*, 2015, pp. 2791–2799.
Michael Jones  Michael Jones received his Ph.D. in 1997 from the Massachusetts Institute of Technology. He was a member of DEC's Cambridge Research Lab from 1997 to 2001 and is currently a senior principal research scientist at Mitsubishi Electric Research Laboratories (MERL) in Cambridge, Massachusetts where he has been since 2001. He has published papers in many areas of computer vision and machine learning and is mainly interested in object detection, action detection, face recognition, person re-identification and video anomaly detection.

Ranga Raju Vatsavai  Raju is a Chancellor’s Faculty Excellence Program Cluster Associate Professor of Geospatial Analytics in the Department of Computer Science, North Carolina State University (NCSU). Before joining NCSU, Raju was the Lead Data Scientist for the Computational Sciences and Engineering Division (CSED) at the Oak Ridge National Laboratory (ORNL). He works at the intersection of spatial and temporal big data management, machine learning, and high-performance computing with applications in the national security, geospatial intelligence, natural resources, agriculture, climate change, location-based services, and human terrain mapping. He holds MS and PhD degrees in computer science from the University of Minnesota.