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 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 and provide a comprehensive comparison of the accuracy of many algorithms on standard test sets. 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.

This survey focuses on single-scene video anomaly detection because it is the most common use case for video anomaly detection in real-world applications. The motivating example is a surveillance camera monitoring a scene and a person responsible for noticing any unusual activity that occurs. This scenario highlights the practical importance of developing algorithms for single-scene video anomaly detection, because this is clearly a task that would be better done by a computer given the extreme difficulty for a person to pay attention to a camera feed (typically with nothing interesting occurring) for long periods of time. If this scenario were changed to a person monitoring a bank of camera feeds, it would still be best modeled as several single-scene video anomaly detection problems for two reasons: (1) the need to handle location-dependent anomalies and (2) the possibility that all the scenes in the camera feeds are not consistent.

Most prior work on video anomaly detection has not recognized the important distinction between single-scene video anomaly detection and multi-scene. One important difference is that the single-scene anomaly detection formulation can contain location-dependent anomalies whereas multi-scene cannot. The lack of recognition of the single-scene/multi-scene distinction is likely due to the fact that most single-scene video anomaly detection datasets (Street Scene being the main exception) contain very few location-dependent anomalies which means that methods that do not accommodate location-dependent anomalies are not heavily penalized. A location-dependent anomaly is an object or activity that is anomalous in some regions of a scene but not in other regions. A good example is walking on grass. In a particular scene, there may be some areas of grass that are normal to walk on and other areas that are restricted and thus anomalous to walk on. The only factor that distinguishes these two activities is the location. In multi-scene video anomaly detection, normal video from many different, unrelated scenes are 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

- *Bharathkumar Ramachandra is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695 USA. E-mail: bramach2@ncsu.edu.*
- *Michael J. Jones is with the Mitsubishi Electric Research Labs (MERL), Cambridge, MA 02139 USA. E-mail: mjones@merl.com.*
- *Ranga Raju Vatsavai is with the Behavioral Reinforcement Learning Lab, Lirio and the Department of Computer Science, North Carolina State University, Raleigh, NC 27695 USA. E-mail: rvatsavai@ncsu.edu.*

Manuscript received 16 Jan. 2020; revised 14 Oct. 2020; accepted 21 Nov. 2020. Date of publication 25 Nov. 2020; date of current version 1 Apr. 2022. (Corresponding authors: Michael J. Jones.) Recommended for acceptance by V. Lepetit. Digital Object Identifier no. 10.1109/TPAMI.2020.3040591

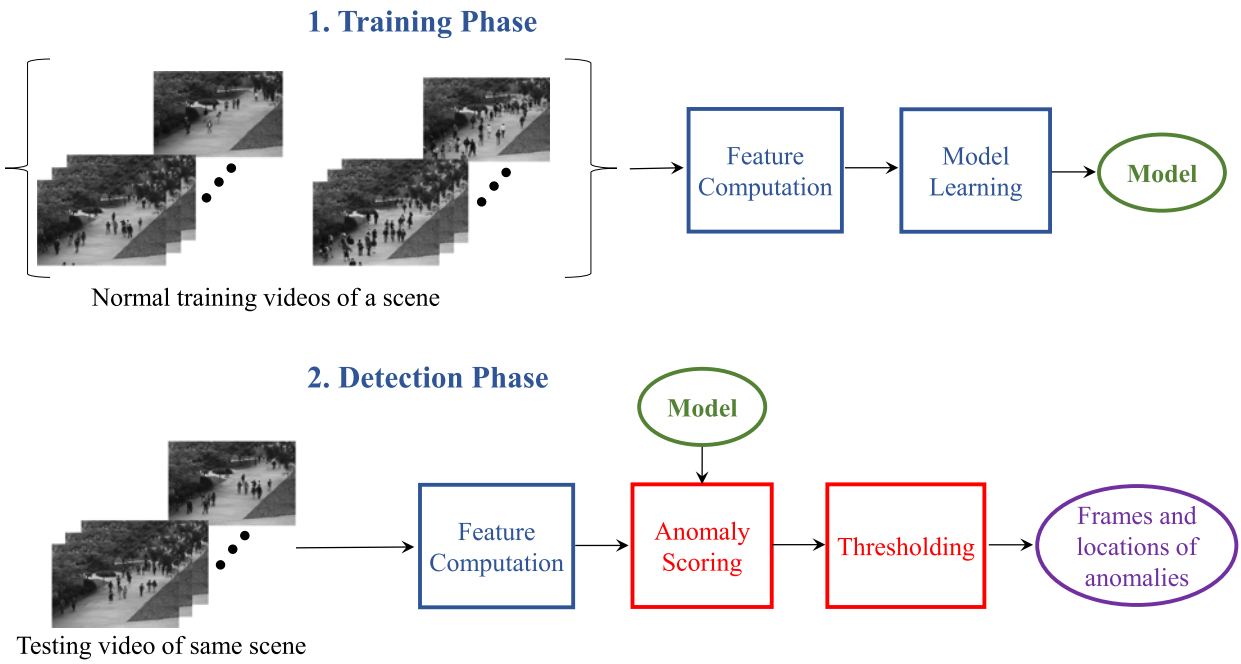

Fig. 1. Overview of single-scene video anomaly detection. Typical algorithms include a model building phase in which a model of normal activity is learned from one or more videos of a scene followed by a detection phase in which anomalies are detected in video from the same scene.

occur *anywhere in any of the videos*. Because there is no correspondence across scenes in the multi-scene formulation it is not possible to create a single model in which an activity is anomalous in some locations of some scenes but not in other locations. 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 or objects occurring in unusual locations are commonplace for single-scene video anomaly detection.

Another important consideration for multi-scene video anomaly detection that does not apply to single-scene is that the normal training videos need to be "consistent" in the sense that what is normal and what is anomalous must be the same in all of the scenes. This is because a single model of normality is being built from video of all the different scenes. For example, a truck backing up to a building may be normal in one scene because there is a loading dock while a truck backing up to a building in another scene may be anomalous. Such scenes would not be consistent.

Because of the practical importance of single-scene video anomaly detection, we focus on this formulation of the problem in this survey.

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.

Fig. 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. We have already discussed the multi-view formulation ([7], [8], [9], [10], [11]) above in some detail.

Another alternate formulation for video anomaly detection that has been used in a number of papers ([12], [13], [14], [15], [16]) 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 [17]. 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 a car skidding sideways on the road. Detecting these anomalies usually only requires 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, a car weaving in and out of traffic, or loitering around foreign embassy buildings. 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 of reference" 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 another possible formulation that some have considered in Section 2.4.3. 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 description of the problem formulation considered 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 ([8], [9], [10], [18]), 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 only that *something* in the frame is anomalous may leave the user wondering exactly what triggered the anomaly detector. We clearly outline which past works focus solely on temporal localization and which include spatial localization as well.

TABLE 1
Characteristics of Video Anomaly Detection Datasets

Dataset	Total Frames	Training Frames	Testing Frames	Anomalous Events	Pixel-wise annotation	Track ID annotation
Subway*	125,475	22,500	102,975	85	N	N
UMN**	3,855	N/A	N/A	11	N	N
UCSD Ped1, Ped2*	18,560	9,350	9,210	77	Y	Y
CUHK Avenue	30,652	15,328	15,324	47	N[1]	Y
Street Scene	203,257	56,847	146,410	205	N	Y

*Aggregates from 2 Scenes.
**Aggregates from 3 Scenes. Adapted from [6].

1.4 Other Surveys

Two past surveys focus on crowded scene analysis ([19], [20]), 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.* [21] focused on anomaly detection in surveillance videos, but is a high-level view of the area, does not cover the most recent work and does not include a comprehensive performance evaluation of different algorithms as our survey does. A short survey by Chong *et al.* [22] from 2015 is narrowly focused on different methods of modeling video and does not include a comparison of methods on video anomaly detection datasets. Finally, a survey by Kiran *et al.* [23] from 2018 focuses mainly on reconstruction approaches to video anomaly detection and also does not provide a comprehensive comparison across many methods in the field. Unlike past surveys, ours includes a discussion and categorization of a broad selection of methods for video anomaly detection, a quantitative comparison of many different algorithms on standard datasets, a discussion of the important publicly available datasets, a discussion of various evaluation criteria, as well as recent trends and directions for future research.

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. In Section 6, we present a comparative study between the various methods. Finally, in Section 7, we discuss the state of research in this field and provide some recommendations for future research directions.

2 Overview of Single-Scene Video Anomaly 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 Fig. 2 shows one normal frame and one frame with a single anomaly for each of the datasets we recommend for use.

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

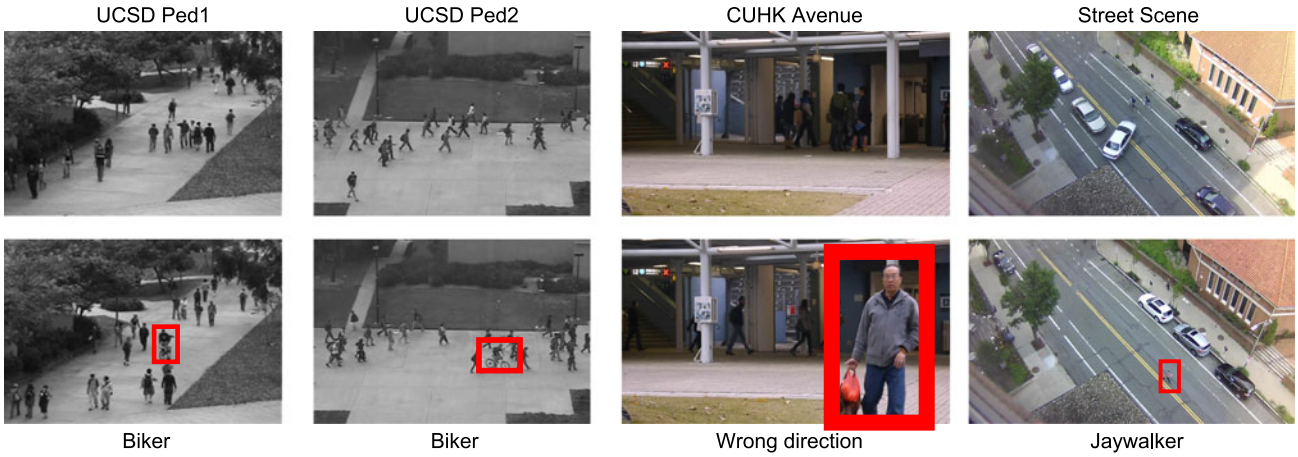

Fig. 2. One normal frame and one frame with an anomaly from each of the recommended datasets for single-scene video anomaly detection.

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 [2], [24]. 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 [25] and made available at https://hci.iwr.uni-heidelberg.de/COMPVIS/research/parsing/. Very recently, the authors in [26] 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 [27] at http://www.merl.com/demos/video-anomaly-detection.

2.1.4 CUHK Avenue

The CUHK Avenue dataset [3] 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. In [28], the authors double the size of the dataset and label spatial locations of the abnormal events. 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".

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 [27] 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], [14], [29], [30]. 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 1280×720 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 a few 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. The dataset is available for download at https://svip-lab.github.io/dataset/campus_dataset.html.

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. The dataset can be downloaded from the project page at https://www.crcv.ucf.edu/projects/real-world/.

Car Dashcam Datasets

Another interesting and large subset of multi-view datasets are dashcam video datasets taken from moving cameras inside of cars and trucks. These include datasets from [31], [32], [33] (called D^2-City), and [34] (called RetroTrucks). Anomalies in these datasets mainly consist of traffic accidents.

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 handholding? 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 despite the inevitable ambiguities in labeling.

2.2.1 Traditional Criteria

Traditionally, research in this field has used frame-level and pixel-level criteria to evaluate performance, first described in [24] (which also presented the UCSD Pedestrian datasets). These criteria basically describe how to count positives, negatives, true positive and false positives and subsequently compute true positive rate (TPR) and false positive rate (FPR) at a given anomaly score threshold

$$\text{TPR} = \frac{\text{num. of true positive frames}}{\text{num. of positive frames}}$$

$$\text{FPR} = \frac{\text{num. of false positive frames}}{\text{num. of negative frames}}.$$

Then, the threshold on anomaly score is varied in order to generate Receiver Operating Characteristic (ROC) curves of FPR versus TPR. Area under the ROC curve and Equal Error Rate (EER) are used to summarize an ROC curve.

These criteria use pixel-level ground truth. That is, every frame at time t, $\mathbf{F^t}$, has a corresponding binary ground truth mask $\mathbf{A^t}$ indicating whether or not each pixel is anomalous.

The frame-level criterion is as follows: Given the predicted per-pixel anomaly score map $\mathbf{S^t}$ corresponding to the tth frame of a test video, the frame is said to be predicted as anomalous if $\sum_p[\mathbf{S^t}(p) \geq \Gamma] \geq 1$ where p indexes over all pixels in a frame and Γ is the anomaly score threshold. The notation $[C]$ evaluates to 1 if condition C is true (or 1) otherwise it evaluates to 0. Further, a frame predicted to be anomalous at time t is counted as a true positive frame if $\sum_p[\mathbf{A^t}(p) == 1] \geq 1$ and as a false positive if $\sum_p[\mathbf{A^t}(p) == 1] == 0$.

In other words, frames are predicted as anomalous if they have at least one pixel that received a score larger than the anomaly score threshold. A frame predicted to be anomalous is counted as a true positive if the annotation for that frame has at least one ground truth anomalous pixel and a false positive otherwise. The total number of positives and negatives are determined by the frame-level annotations and are used to compute TPR and FPR, and subsequently AUC and EER.

$$\text{num. of positive frames} = \sum_{t=1}^{T}(\sum_p[\mathbf{A^t}(p) == 1] \geq 1)$$

$$\text{num. of negative frames} = \sum_{t=1}^{T}(\sum_p[\mathbf{A^t}(p) == 1] == 0),$$

where t indexes over testing frames and T is the total number of testing frames. In other words, the number of positive frames is the number of testing frames with at least one ground truth anomalous pixel while the number of negative frames is the number of testing frames with no ground truth anomalous pixels.

The frame-level criterion does not evaluate whether any spatial localization has been achieved and the authors themselves recommended against using solely this criterion in [2], instead suggesting use of the pixel-level criterion.

The pixel-level criterion is as follows: Given the predicted anomaly score map $\mathbf{S^t}$ corresponding to the tth frame of a test video, the frame is counted as a true positive frame if $\sum_p[(\mathbf{S^t}(p) \geq \Gamma) \cdot \mathbf{A^t}(p)] \geq 0.4 \cdot \sum_p[\mathbf{A^t}(p) == 1]$ and $\sum_p[\mathbf{A^t}(p) == 1] \geq 1$. Conversely, the frame is counted as a false

positive frame if $\sum_p[\mathbf{S^t}(p) \geq \Gamma] \geq 1$ and $\sum_p[\mathbf{A^t}(p) == 1] == 0$.

In other words, a frame is counted as a true positive frame if over 40 percent of the annotated ground truth anomalous pixels in a frame are predicted as anomalous by the model. If a frame has no ground truth anomalous pixels and yet even one pixel is predicted as anomalous, 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 percent threshold is applied over *all ground truth anomalous pixels in a frame*.
2) A frame that contains at least one ground truth anomalous pixel cannot count as a false positive regardless of any incorrect regions in the frame that are predicted as anomalous.
3) A frame without any ground truth anomalous pixels can be counted for only one false positive even if there are multiple distinct regions that are predicted as anomalous.
4) The criterion does not penalize looseness of a predicted region. That is, as long as 40 percent of annotated pixels are predicted as anomalous, it does not hurt performance to change the prediction mask to the entire frame.

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 prediction masks with a filter of the same size as the frame (i.e., if a single pixel is predicted as anomalous in a frame, make all pixels in the frame anomalous). At a given threshold, this can *only increase the true positive rate without changing the false positive rate* according to the pixel-level criterion.

We should also note that in [24], the authors fail to fully describe pixel-level evaluation measure. Specifically, the authors define a true positive as a frame where at least 40 percent of the truly anomalous pixels in the frame are predicted as anomalous, 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 percent of the pixels are predicted as anomalous 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-level AUCs being reported.

Although these criteria, if correctly used, 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 [35] proposed the *Dual Pixel Level* criterion which adds an additional constraint to the pixel-level criterion. With the same notation as before, a frame at time t would only be counted as a true positive if $\sum_p[(\mathbf{S^t}(p) \geq \Gamma) \cdot \mathbf{A^t}(p)] \geq 0.4 \cdot \sum_p[\mathbf{A^t}(p) == 1]$ and $\sum_p[(\mathbf{S^t}(p) \geq \Gamma) \cdot \mathbf{A^t}(p)] \geq 0.1 \cdot \sum_p[\mathbf{S^t}(p) \geq \Gamma]$ and $\sum_p[\mathbf{A^t}(p) == 1] \geq 1$.

That is, in addition to the pixels predicted as anomalous needing to cover at least 40 percent of the ground truth anomalous pixels, at least 10 percent of the pixels predicted as anomalous also need to be *covered by* the ground truth anomalous pixels. In other words, the pixels predicted as anomalous 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 ground truth anomalies, (b) both true positive as well as false positive predicted pixels/regions and (c) multiple false positive predicted pixels/regions. The authors of [28] 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. In their perspective, 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 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 detected region 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.

These criteria require reasoning about ground truth and detected anomalous regions within frames as opposed to whole frames. Some annotated datasets specify ground truth as bounding boxes which directly give the ground truth regions. For datasets that specify ground truth as pixelwise masks, a set of anomalous regions can be computed as connected components of anomalous pixels. Similarly, detected regions can be computed as connected components of detected pixels for algorithms that return pixelwise anomaly masks.

The *region-based detection criterion* calculates the region-based detection rate (RBDR) over all frames in the test set versus the number of false positive regions per frame.

$$\text{RBDR} = \frac{\text{num. of true positive regions (NTP)}}{\text{total num. of anomalous regions (TAR)}}. \tag{1}$$

The RBDR is computed over all ground truth anomalous regions in all frames of the test set.

$$\text{FPR} = \frac{\text{num. of false positive regions (NFP)}}{\text{total frames}}, \tag{2}$$

where FPR is the false-positive rate per frame.

The number of true positive regions (NTP) can be expressed as

$$\text{NTP} = \sum_{t=1}^{T}\sum_{i=1}^{N_t}[\exists\ D^t \text{ such that } \frac{G_i^t \cap D^t}{G_i^t \cup D^t} \geq \beta], \tag{3}$$

where D^t is a detected region in frame t, G_i^t is the ith ground truth anomalous region in frame t, N_t is the number of ground truth anomalous regions in frame t, and β is a threshold which is set to 0.1 in [6].

In other words, the number of true positive regions is the total number of ground truth regions in all testing frames that are detected. A ground truth region in a frame is considered detected if the intersection over union between the ground truth region and any detected region in the frame is greater than or equal to β.

The total number of ground truth anomalous regions can be expressed as

$$\text{TAR} = \sum_{t=1}^{T} N_t, \tag{4}$$

where N_t is the number of ground truth anomalous regions in frame t.

The number of false positive regions (NFP) can be expressed as

$$\text{NFP} = \sum_{t}^{T}\sum_{j=1}^{M_t}[\forall\ G^t, \frac{G^t \cap D_j^t}{G^t \cup D_j^t} < \beta], \tag{5}$$

where G^t is a ground truth anomalous region in frame t, D_j^t is the jth detected region in frame t, M_t is the number of detected regions in frame t, and β is a threshold set to 0.1.

In other words, the number of false positive regions is the total number of detected regions in all testing frames that do not overlap enough with any ground truth anomalous region.

The other criteria introduced in [6], the *track-based detection criterion*, measures the track-based detection rate (TBDR) versus the number of false positive regions per frame. For this criterion, ground truth anomalous tracks are needed. A ground truth anomalous track is a set of ground truth anomalous regions in a sequence of consecutive frames.

$$\text{TBDR} = \frac{\text{num. of true positive tracks (NTPT)}}{\text{total num. of anomalous tracks (NAT)}}. \tag{6}$$

Without loss of generality, let us assume that in the notation G_k^t, k further identifies an anomalous track. Then, an anomalous track can be defined as the set of ground truth anomalous regions it contains, spanning frames $t1$ to $t2$ as such

$$L_k := \{G_k^{t1}, G_k^{t1+1}, \ldots, G_k^{t2-1}, G_k^{t2}\}.$$

The number of true positive tracks can be expressed as

$$\text{NTPT} = \sum_{k=1}^{N_k}\left[\left(\sum_{G_k^t \in L_k}[\exists\ D^t \text{ such that } \frac{G_k^t \cap D^t}{G_k^t \cup D^t} \geq \beta]\right) \geq \alpha \cdot |L_k|\right], \tag{7}$$

where N_k is the total number of anomalous tracks (NAT), $|L_k|$ denotes the the size of L_k and α is a threshold which is set to 0.1 in [6].

In other words, a ground truth anomalous track is considered a true positive if at least a fraction α (set to 0.1) of the ground truth anomalous regions in the track are correctly detected. The condition for detecting ground truth anomalous regions is the same as for the region-based criterion above.

$$\text{FPR} = \frac{\text{num. of false positive regions (NFP)}}{\text{total frames}}, \tag{8}$$

where FPR is the false-positive rate per frame. A region predicted as anomalous in a frame is a false positive if the IOU between it and every ground truth region in that frame is less than β. This is the same definition as for the region-based criterion.

Notice that since false positive regions are counted per frame, 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 [36]. For these reasons, researchers are encouraged to provide qualitiative 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 Fig. 3 for intuition on how these

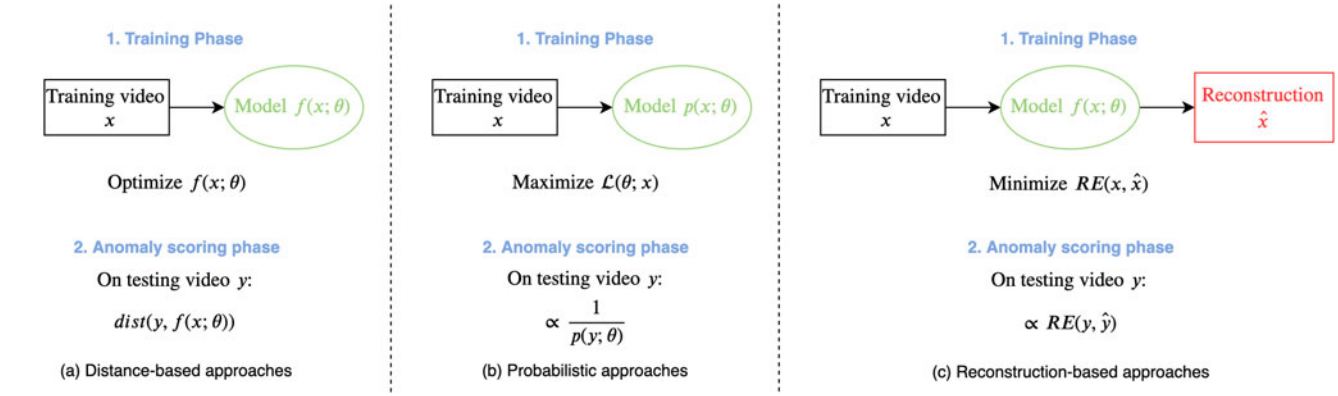

Fig. 3. An overview of the 3 basic approaches past work has taken to video anomaly detection.

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 Fig. 1, 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], [30]), dynamic textures ([2], [24]), histogram of gradients ([10], [37], [38]), histogram of flows ([10], [37], [39], [40]), flow fields ([4], [25], [41], [42], [43]), dense trajectories ([38]) and foreground masks ([6], [25]). The deep features are further either extracted as-is from a pre-trained network (such as [29], [30], [44], [45], [46], [47]) 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 [10], [11], [18], [35], [48], [49], [50]).

Another consideration in representation is the atomic unit of processing. Algorithms process atomic units ranging from image patches (such as [25], [39], [48]) to video patches (such as [2], [3], [4], [6], [11], [24], [27], [30], [35], [37], [40], [43], [46], [49], [51], [52], [53]) to single full frames (such as [29], [44], [54]) and even video snippets (short sequences of full frames) (such as [9], [10], [18], [38], [42], [45], [47]). When dealing with image or video patches, algorithms operate on units from single fixed-size patches (such as [4], [6], [37]) to multi-scale fixed-size patches (such as [25], [41], [48]) to arbitrarily-sized region proposals (such as [55]).

2.3.2 Broad Themes in Modeling

Broad themes in modeling include the use of one-class SVMs (such as [29], [30], [38], [48]), nearest neighbor approaches (such as [1], [6], [27], [40], [44]), Hidden Markov Models (such as [2], [51], [52], [53]) and more generally Probabilistic Graphical Models (such as [2], [25], [41]). More recently, deep learning approaches have started using adversarial training strategies (such as [9], [50], [54]).

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 ([8], [10], [22]). 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 [9], [10], [11], [18], [39], [46], [48], [54]. 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], [6], [27], [37]), (2) providing additional context in the form of information from neighboring voxels for scoring (such as [2], [41], [51]).

Another problematic practice that has emerged is that of *per-video normalization*, such as that in [9], [10], [18], [46], [54], [56], [57]. 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 Less Common Settings and Related Modeling

2.4.1 Object Detection and Tracking Approach

This approach relies on being able to detect and track objects across time, creating complete object trajectories. Unfortunately, natural scenes rarely have the property that only certain object types that can be detected, such as humans ([7], [58], [59]) or cars, are present. Moreover, natural scenes almost always present with occlusions. So while object detection and tracking methods would have vast utility in anomaly detection

with the ability to detect trajectory anomalies, as of the current state of the art in computer vision, this is a clearly suboptimal approach for general video anomaly detection.

For a specific example, consider the work of Morais *et al.* [7]. The authors take a human-detection-and-tracking reconstruction-based approach. They extract fixed-length tracklets for skeletons estimated by Alpha Pose [60] and decompose them into global and local components based on their properties. They design a global + local two branch architecture, each of which contains three GRUs [61] - an encoder, a reconstructing decoder and a predicting decoder. The global and local branches pass messages to each other about their states and the whole architecture is jointly optimized for reconstruction and future movement prediction through mean squared error terms on perceptual, global and local loss terms. While effective for detecting anomalous human actions, their method is not applicable to the general video anomaly detection problem. In their experiments, they exclude those sequences from ShanghaiTech and CUHK Avenue where the main subject is non-human or cannot be detected and tracked.

2.4.2 Supervised Anomaly Detection

Supervised approaches assume 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 and (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 [62]. 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 [63] 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 [64], 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 [65] 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 representation and formulation lead to good performance. From Fig. 3a, distance-based approaches can be seen as a more general form of both probabilistic and reconstruction-based approaches.

Many different types of features have been used in distance-based approaches as well as many different ways of measuring distance to the normal features. One common approach for many methods is to use one-class SVMs to compute a decision boundary around feature vectors from normal training video (such as [11], [29], [30], [38], [48], [66]). A disadvantage of such approaches is that it is expensive to update the model given new normal training data. The SVM learning algorithm must be rerun on all of the old plus new data. An alternative approach is to use a mixture of Gaussians to model normal feature vectors and then Mahalanobis distance to measure distance to normality. This idea has been used in the works [35], [47], [49], [67].

In terms of the features used to represent video volumes, early approaches used a variety of hand-crafted features including foreground masks [6], [37], histograms of flow [68], motion magnitude [37], histogram of gradients [69], motion boundary histograms [70], dense trajectories [71], and space time interest point (STIP) features [72]. More recent approaches in [6], [11], [29], [30], [45], [47], [48], [66] have focused on features learned by deep networks which generally have higher accuracy (see Table 3). These deep-network-based methods encompass a variety of ways of learning deep features and a variety of ways of using the deep features in different models of normality.

The remainder of this section briefly summarizes a number of distance-based approaches.

In [37], the main premise is 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 volumes and represent them with low-level motion descriptors. Aggregated K-nearest neighbors (K-NN) distances between normal and test video volumes are used to compute anomaly scores.

In [38], the authors extract a set of social force [73], HOG (histogram of gradients) [69], HOF (histogram of optical

flow) [68], motion boundary histogram (MBH) features [70] and dense trajectories [71] from video snippets. They use *Vector Quantization (VQ) coding* to represent the features and a *one-class SVM* [74], with either linear, RBF or polynomial kernels to perform anomaly detection.

In [48], 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 volumes to learn low-dimensional representations. They then use the latent codes from the DAEs in a one-class SVM [74] 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 [49], the authors focus on a fast method for video anomaly detection. They use *Gaussians to model the distributions of features* from a simple 2-layer auto-encoder as well as the distribution of distances of each video volume to its spatio-temporal neighbors using a Structural Similarity Index Measure (SSIM). They detect anomalies by computing the Mahalanobis distances to the training Gaussians.

Sabokrou *et al.* [35] build on their earlier work in [49]. They use many internal layers of a 3D auto-encoder as well as a deep CNN to provide features which are modeled by Gaussian distributions in a cascade structure.

This work was followed by [47] which simplifies the cascade architecture into a two-layer cascade of Gaussian models and uses features from a pre-trained fully convolutional network. The resulting algorithm can process hundreds of video frames per second on a high-end GPU.

Smeureanu *et al.* [29] present *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 [74] with a linear kernel on deep features extracted from a VGG-f network on each mean-subtracted frame [75]. They smooth their score maps with a spatio-temporal filter and perform localization by dividing video into fixed-size video volumes and simply aggregating anomaly scores over the patch regions.

In [66], the authors present a way to use *convolutional winner-take-all auto-encoders* [76] to learn motion-feature representations from optical flow fields of fixed-size video volumes. They then use the learnt motion-feature representations to build location-dependent one-class SVMs [74] to perform anomaly scoring.

In [77], 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 [78], 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 [67], the authors build a model of normality using the *Growing Neural Gas* [79] algorithm on STIP features [72] extracted from video snippets/volumes. 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.

In [45], the authors present another way to use image features from a pre-trained convolutional network, AlexNet [80]. 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* [81] 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 [50], the authors present *one of the first approaches to use adversarial training* for video anomaly detection. They use a discriminator network ($\mathcal{D}$) tasked with distinguishing original image patches from reconstructions of noisy patches obtained from a denoising auto-encoder network ($\mathcal{R}$) which plays the role of generator. Since $\mathcal{R}$ is trained only on image patches from training data, it decimates outliers and thus enables $\mathcal{D}$ to tell an anomalous image patch from its reconstruction easily.

In [30], the authors propose a two-stage anomaly detection algorithm. They extract fixed-size video volumes 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 [30]. They use feature pyramid networks [82] 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 recently released dataset, Street Scene. They use a simple *nearest neighbor location-dependent anomaly detection* scheme using hand-crafted representations of video volumes (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 [27], 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* [83]. The Siamese network is trained to classify video patch pairs as similar or different and is used to find testing video volumes that are different from all training video volumes and are therefore anomalous. An exemplar model (consisting of all unique normal video volumes) is learned from training data of the target dataset. Finally, nearest neighbor scoring between test video volumes 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 Fig. 3b for this intuition. Most of the probabilistic approaches came before the wave of deep learning methods and instead rely on features such as spatio-temporal gradients [53], optical flow fields [4], [25], [41], [42], [43], [51], and STIP features [40] coupled with traditional models such as Markov Random Fields [51], [52], [53] and mixtures of Gaussians [42], [53]. A couple of more recent approaches do make use of deep networks [44], [84] and also show improved accuracy. These approaches enjoy a favorable property in being highly principled and having the ability to model the continuous nature of anomalousness well. Unfortunately, they are often very slow at test time. We summarize the various probabilistic approaches below.

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 [43], the authors utilize the *social force* model [73]. Optical flow is used 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. A bag-of-words model is used to model social force interactions and anomalies are detected as low-likelihood frames under the model.

In [52], 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 [53], 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.

Kim *et al.* [51] present 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) [85] 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 [42], the authors also advect particles on a grid of optical flow observations on video similar to [43], 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 [24], the authors propose learning a *Mixture of Dynamic Textures* (MDT) [86], [87] 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 [88]. 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 [89] framework.

The authors in [25] 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 [41], they further build on this idea by considering object hypotheses in the form of *flexible video pipes instead of just image patches*.

In [40], the authors propose a hierarchical local plus global method to detect anomalies. They model video with Spatio-Temporal Interest Point features [72] and form a codebook with K-means clustering, detecting local anomalies as those with high distance to the kth 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 [90] with an RBF kernel for each model. They then designate low-likelihood test ensembles under the kth nearest neighboring GPR ensemble model as anomalous.

In [44], the authors propose a *unique method to recount anomalous events as they are detected*. They first train a Fast-RCNN [91] model to predict object, action and attribute classes from large-scale COCO [92] and Visual Genome [93] 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 [84], the authors use PCANet [94] to extract deep representations, learned from 3D gradients of normal image patches. They then use *Deep GMMs* [95] *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 represent the input (images or video snippets) using a high-level or compact representation learned from normal video and then reconstruct the input using only this representation. They are based on the premise that out-of-distribution inputs such as anomalies are inherently harder to reconstruct using a representation learned from normal video when compared to in-distribution normal data, thus justifying the use of reconstruction error as a proxy for anomaly score. See Fig. 3c for an illustration of this intuition. Almost all of the reconstruction-based approaches use modern deep learning methods, and in particular, most are based on either convolutional auto-encoders [10], [18], [26], [56] or generative adversarial networks (GANs) [26], [54], [96], [97]. Generally, reconstruction-based approaches have the disadvantage that the models they use (e.g., auto-encoder or GAN) need to be retrained to accomodate new normal training video. Many of these approaches do not evaluate spatial localization of anomalies despite the fact that reconstruction error is generally pixelwise. Presumably, this is because their spatial localization accuracy is low. Another disadvantage of auto-encoder-type reconstruction methods is that reconstruction errors for frames are proportional to the number of foreground objects in the frame and this is the reason most of these methods have to employ a post-processing step of per-video normalization, as previously discussed. We summarize the various reconstruction-based approaches in the remainder of this section.

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. Rather than demonstrating robustness of features, we believe this actually indicates a common bias towards anomalous activities being caused by objects with faster motion in the group of datasets they perform their experiments on.

Chong *et al.* [18] build on the convolutional auto-encoder architecture of [10] by preserving temporal ordering of frames through the convolutions and modeling the temporal information at the bottleneck layer with specialized *convolutional LSTM* [98] layers.

In [54], the authors attempt the *first use of Generative Adversarial Networks* [99] 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, and use per-video normalization to perform anomaly scoring for detection and pixel-wise localization.

In [100], the authors perform feature learning and reconstruction on fixed-size raw video patches using *Restricted Boltzmann Machines* (RBMs) [101] using the Contrastive Divergence [102] 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 [103] that takes training video snippets of length t as input and predicts a future frame for time $t + 1$. Further, they use FlowNet [104] 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 devoncolutional 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. 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 [26], the authors observe 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 [54]. 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 [103] 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 [105].

In [57], the authors contend that CNN-based reconstruction approaches suffer from reconstructing anomalous events well because of CNNs' high representational capacity. They

TABLE 2
Thematic Grouping by Representation and Modeling Strategies Taken

Method	Approach	Representation theme			Modeling theme			
		Proc. unit	Input feats.	Pre-trained net	Model component	Per-video norm.	Location-dependent	Sp-local.
[37]	Dist.	Fixed-size VP	HOF, fg%, motion magnitude		NN		✓	
[38]	Dist.	VS	HOF, SF, dense trajectories		OC-SVM			
[48]	Dist.	Fixed-size IP	Raw, flow, deep		OC-SVM, AE			✓
[49]	Dist.	Fixed-size VP	Raw, SSIM, deep		AE			✓
[35]	Dist.	Fixed-size VP	Raw, deep		AE		✓	✓
[29]	Dist.	FF	Raw, deep	✓	OC-SVM			✓
[66]	Dist.	Fixed-size VP	Flow, deep		OC-SVM, AE		✓	✓
[77]	Dist.	FF	dense trajectories					✓
[67]	Dist.	VS, VP	STIP		NN			✓
[47]	Dist.	VS	Raw, deep	✓	AE			✓
[45]	Dist.	FF, VS	Flow, deep	✓			✓	✓
[50]	Dist.	Fixed-size IP	Raw, deep		Adversarial, AE			
[30]	Dist.	Fixed-size VP	3D grad., deep	✓	OC-SVM		✓	✓
[11]	Dist.	Fixed-size VP	2D grad., deep		SVM, AE			
[6]	Dist.	Fixed-size VP	Flow, fg-mask		NN		✓	✓
[27]	Dist.	Fixed-size VP	Flow, deep		NN		✓	✓
[4]	Prob.	Fixed-size VP	Flow				✓	✓
[43]	Prob.	Fixed-size VP	Flow, social force					✓
[52]	Prob.	Fixed-size VP	Fg-mask, co-occurrence matrix		HMM		✓	✓
[53]	Prob.	Fixed-size VP	3D grad.		HMM		✓	✓
[51]	Prob.	Fixed-size VP	Flow		HMM		✓	✓
[42]	Prob.	VS	Flow				✓	✓
[24]	Prob.	Fixed-size VP	DT				✓	✓
[25]	Prob.	Fixed-size IP	Fg-mask, flow		OC-SVM		✓	✓
[2]	Prob.	Fixed-size VP	DT		HMM		✓	✓
[41]	Prob.	Fixed-size VT	Fg-mask, flow		OC-SVM		✓	✓
[40]	Prob.	Fixed-size VP	STIP, 3DSIFT, HOG, HOF		NN		✓	✓
[106]	Prob.	Fixed-size VP	3D grad., HOF		OC-SVM		✓	✓
[44]	Prob.	FF	Raw, deep	✓	NN, OC-SVM		✓	✓
[84]	Prob.	Fixed-size IP	3D grad., deep					✓
[10]	Recon.	VS	Raw, deep		AE	✓		
[22]	Recon.	VS	Raw, deep		AE	✓		
[54]	Recon.	FF	Raw, flow, deep		Adversarial	✓		✓
[100]	Recon.	Fixed-size VP	Raw, deep				✓	✓
[9]	Recon.	VS	Raw, flow, deep, 2D grad.		Adversarial	✓		
[96]	Recon.	FF	Raw, flow, deep		Adversarial, AE	✓		✓
[26]	Recon.	VS	Raw, flow, deep		Adversarial, AE			✓
[97]	Recon.	VS	Raw, flow, deep		Adversarial			✓
[57]	Recon.	VS	Raw, deep		AE	✓		
[39]	S-Recon.	IP, VP, VS	HOF. flow					✓
[3]	S-Recon.	Fixed-size VP	3D grad.				✓	✓
[46]	S-Recon.	Fixed-size VP	Deep	✓		✓		
[56]	S-Recon.	VS	Raw, deep	✓	AE	✓		✓

propose augmenting a U-net style encoder-decoder future frame prediction/reconstruction network with a *learned memory module* that stores important normal patterns and computing anomaly scores using a combination of PSNR between a frame and its reconstruction as well as distance between an encoding and nearest memory element. They also perform per-video normalization of scores.

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. Almost all sparse reconstruction approaches optimize a sparse combination learning formulation of some kind [3], [28], [39]. These approaches usually enjoy some favorable properties in being fast (since sparsity is a goal) and having models of normality that are easy to update in an online fashion. A disadvantage of these approaches is that they often rely too heavily on *memorizing* salient normal features, placing a large burden on the normal training set being exhaustive. They also do not tend to model the intuition behind the continuous nature of anomalousness very well due to this, that is, anomalies that received smaller scores often do not necessarily correspond to less anomalous activities as per human intuition.

In [39], the authors estimate optical flow fields in video and extract a multi-scale histogram of flow features. They then learn a *dictionary* of these features from training video and use a *sparse reconstruction cost* based on L1 minimization as their anomaly score on volumes from test video. Favorable properties include online update of the dictionary and the ability to define bases over different representations such as image patches, video volumes 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 [28], the authors extend this work into a *birth-and-death combination online solver* to handle both dynamic and large-scale data. They also improve detection speed from 150 FPS to an impressive 1,000 FPS.

In [46], the authors propose a temporally-coherent sparse coding (TSC) approach *with the constraint that temporally close frames be encoded with similar sparse coefficients*. They use a special type of stacked recurrent neural network (sRNN) to enforce this and by optimizing all parameters of this network simultaneously, avoid the non-trivial hyperparameter 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.

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, Complied From This Same List

Method	UCSD Ped1 frame AUC/EER	UCSD Ped1 pixel AUC*	UCSD Ped2 frame AUC/EER	UCSD Ped2 pixel AUC	CUHK Avenue frame AUC/EER
Adam [4]	65.0%/38.0%	46.1%	63.0%/42.0%	18.0%	-/-
Social force [43]	67.5%/31.0%	19.7%	63.0%/42.0%	21.0%	-/-
MPPCA [24]	59.0%/40.0%	20.5%	77.0%/30.0%	14.0%	-/-
Social force + MPPCA [24]	67.0%/32.0%	21.3%	71.0%/36.0%	21.0%	-/-
MDT [24]	81.8%/25.0%	44.1%	85.0%/25.0%	44.0%	-/-
Video parsing [25]	91.0%/18.0%	83.6%	92.0%/14.0%	76.0%	-/-
Local statistical aggregates [37]	92.7%/16.0%	-	-/-	-	-/-
Detection at 150 FPS (SCL) [3]	91.8%/15.0%	63.8%	-/-	-	-/-
Sparse reconstruction [39]	86.0%/19.0%	45.3%	-/-	-	-/-
HMDT CRF [2]	-/17.8%	82.7%	-/18.5%	-	-/-
AMDN [48]	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 [106]	85.0%/-	65.0%	90.0%/-	-	-/-
Conv-AE [10]	81.0%/27.9%	-	90.0%/21.7%	-	70.2%/25.1%
Deep event models [84]	92.5%/15.1%	69.9%	-/-	-	-/-
Compact feature sets [108]	82.0%/21.1%	57.0%	84.0%/19.2%	-	-/-
Conv-WTA-AE [66]	91.9%/15.9%	68.7%	92.8%/11.2%	80.9%	82.1%/24.2%
RBM [100]	70.3%/35.4%	48.9%	86.4%/16.5%	72.1%	78.8%/27.2%
Convex polytope ensembles [77]	78.2%/24.0%	62.2%	80.7%/19.0%	75.7%	-/-
Joint detection and recounting [44]	-/-	-	92.2%/13.9%	89.1%	-/-
Sparse coding revisit [46]	-/-	-	92.2%/-	-	81.7%/-
GAN [54]	97.4%/8.0%	70.3%	93.5%/14.0%	-	-/-
Online-GNG [67]	93.8%/-	65.1%	94.0%/-	-	-/-
Future frame prediction [9]	83.1%/-	-	95.4%/-	-	85.1%/-
Plug and play CNN [45]	95.7%/8.0%	64.5%	88.4%/18.0%	-	-/-
Fast SCL [28]	93.8%/14.0%	84.1%	95.0%/-	80.0%	-/-
Narrowed normality clusters[30]	-/-	-	-/-	-	88.9%/-
Object-centric auto-encoders [11]	-/-	-	97.8%/-	-	90.4%/-
Appearance-motion cGAN [96]	-/-	-	96.2%/-	-	86.9%/-
$MLAD_{0+3}$ [26]	82.3%/23.5%	66.6%	99.2%/2.5%	97.2%	71.5%/36.4%
Memory-augmented AE [56]	-/-	-	94.1%/-	-	83.3%/-
Prediction+reconstruction [97]	82.6%/-	78.4%	96.2%/-	93.1%	83.7%/-
NN on video patch FG masks [6]	77.3%/25.9%	69.3%	88.3%/18.9%	83.9%	72.0%/33.0%
Siamese distance learning [27]	86.0%/23.3%	80.4%	94.0%/14.1%	93.0%	87.2%/18.8%
Memory-guided normality [57]	-/-	-	97.0%/-	-	88.5%/-

Some of the earlier works unfortunately use only a partially annotated subset available at the time to report performance.

Because they have a focus on temporal anomaly detection, their method does not perform localization.

In [56], 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 use 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 A Comparative Study of Methods

Table 2 lists 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.
- 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 [69].
- HOF: Histogram of Optical Flow [68].
- MBH: Motion Boundary Histogram [70].
- Dense trajectories: [71].
- Social Force: [73].
- DT: Mixtures of Dynamic Textures [86], [87].
- STIP: Spatio-temporal Interest Point features [72].
- 3DSIFT: 3-dimensional Scale Invariant Feature Transform features [107].
- OC-SVM: Use of one-class SVM [74].
- NN: Use of nearest neighbor logic.

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	track AUC			region AUC		
	Ped1	Ped2	Avenue	Ped1	Ped2	Avenue
[6] (FG masks)	84.6%	80.5%	80.9%	46.6%	62.5%	35.8%
[6] (Flow)	86.5%	83.2%	78.4%	48.3%	55.0%	27.3%
[27] (Siamese net)	90.0%	89.3%	78.6%	59.2%	74.0%	41.2%

- 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, local spatial context is considered when detecting anomalies. See Section 1.
- Sp-local.: The method is apparently able to perform spatial localization.

To compare various methods in terms of accuracy, we have compiled Tables 3, 4 and 5 showing the accuracy of many algorithms on the various datasets and evaluation criteria discussed earlier. Although some datasets have evolved both in terms of size and annotations over a period of time, these changes have been modest and we believe that since these datasets are released with pre-defined training and testing splits, the numbers in these tables are a reliable measure of performances in the ways the evaluation criteria aim to capture them. Table 3 compares different methods by their frame and pixel-level criteria scores on UCSD Ped1, UCSD Ped2 and CUHK Avenue datasets. It is arranged chronologically by publication date. Table 4 compares methods via track and region-based criteria for these datasets, highlighting that we may not be as close to saturating performance on these datasets as the traditional criteria might indicate. Table 5 brings into perspective how some of the methods that perform very well on the traditional criteria on the small datasets display very poor performance on the large and complex Street Scene dataset, regardless of the evaluation criteria used.

Something that Table 3 makes clear is that there currently is not a single best method. The method that is best for one dataset and criterion is not the best for a different dataset and criterion. The methods that have the highest accuracy on UCSD Ped1 using the pixel-level criterion, for example, have about middle-of-the-pack accuracy on UCSD Ped2 with the pixel-level criterion.

TABLE 5
Track-Based, Region-Based, Pixel-Level, and Frame-Level Area Under the ROC Curve on Street Scene

Method	track AUC	region AUC	pixel AUC	frame AUC
[10] (Autoencoder)	2%	0.3%	0.1%	61%
[3] (Dictionary method)	10%	2%	7%	48%
[6] (Flow)	52%	11%	17%	51%
[6] (FG masks)	53%	21%	30%	61%

TABLE 6
Running Times of Methods From Literature, Compiled From This Same List

Method	Approach	Fps	Dataset
[49]	Dist.	200	UCSD Ped1, UCSD Ped2, UMN
[29]	Dist.	20	CUHK Avenue
[35]	Dist.	130	UCSD Ped1, UCSD Ped2, UMN
[47]	Dist.	370	UCSD Ped2
[30]	Dist.	24	CUHK Avenue, Subway, UMN
[11]	Dist.	11	CUHK Avenue, UCSD Ped2, ShanghaiTech, UMN
[24]	Prob.	0.4	UCSD Ped2
[25]	Prob.	0.13	UCSD Ped1
[2]	Prob.	1.25	UCSD Ped2
[41]	Prob.	1	UCSD Ped1, UCSD Ped2
[40]	Prob.	2	UCSD Ped1
[18]	Recon.	143	CUHK Avenue, Subway, UCSD Ped1, UCSD Ped2
[9]	Recon.	25	CUHK Avenue
[3]	S-Recon.	150	CUHK Avenue
[39]	S-Recon.	0.26	UCSD Ped1
[46]	S-Recon.	50	UCSD Ped2
[56]	S-Recon.	38	UCSD Ped2
[28]	S-Recon.	1000	UCSD Ped2, CUHK Avenue, Subway

A combination of lack of realistic datasets and evaluation criteria has meant that progress in research in video anomaly detection has not directly translated to high-performing systems deployed in practice. Partly for this reason, reporting running times has not been common practice for research in this field. Nevertheless, Table 6 lists running times during inference for various methods where available. Since the resolution of the frame directly affects processing time for most methods, we also list the datasets on which the runtimes are reported where available. Because different methods used different processors (CPU and/or GPU), the numbers in the table are not directly comparable and should only be used to get a rough idea of the speed of an algorithm. From the table, there is a clear trend where probabilistic approaches, although they can detect anomalies in a very principled framework, struggle to perform detection in real-time.

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. 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. As the field matures into producing approaches that are viable in practice, researchers should also provide runtime analyses of their methods. A qualitative evaluation of quality of false positives is also important, especially to discover biases in modeling. 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 motion magnitudes. 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 [109].

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. Of note is one recent work by Rodrigues *et al*. [58] that has attempted to address loitering anomalies in specific by modeling activities in a multi-timescale manner. 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 modeling strategies that can achieve high detection rates at low false positive rates to make algorithms that are practical for real applications.

REFERENCES

[1] V. Saligrama, J. Konrad, and P.-M. Jodoin, "Video anomaly identification," *IEEE Signal Process. Mag.*, vol. 27, no. 5, pp. 18–33, Sep. 2010.

[2] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, 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 *Proc. IEEE Int. Conf. Comput. Vis.*, 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 Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 555–560, Mar. 2008.

[5] Unusual crowd activity dataset of University of Minnesota, 2006. [Online]. Available: http://mha.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 *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2558–2567.

[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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 996–12 004.

[8] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6479–6488.

[9] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection–A new baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 733–742.

[11] R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7842–7851.

[12] A. Del Giorno, J. A. Bagnell, and M. Hebert, "A discriminative framework for anomaly detection in large videos," in *Proc. Eur. Conf. Comput. Vis.*, 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3313–3320.

[14] R. T. Ionescu, S. Smeureanu, B. Alexe, and M. Popescu, "Unmasking the abnormal events in video," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2914–2922.

[15] Y. Liu, C.-L. Li, and B. Póczos, "Classifier two sample test for video anomaly detections," in *Proc. Brit. Mach. Vis. Conf.*, 2018, Art. no. 71.

[16] G. Pang, C. Yan, C. Shen, A. V. D. Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12 173–12 182.

[17] E. Keogh, J. Lin, and A. Fu, "Hot sax: Efficiently finding the most unusual time series subsequence," in *Proc. 5th IEEE Int. Conf. Data Mining*, 2005, pp. 226–233.

[18] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *Proc. Int. Symp. Neural Netw.*, 2017, pp. 189–196.

[19] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, "Crowd analysis: A survey," *Mach. Vis. Appl.*, vol. 19, no. 5/6, pp. 345–357, Oct. 2008.

[20] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, "Crowded scene analysis: A survey," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 367–386, Mar. 2015.

[21] A. Sodemann, M. Ross, and B. Borghetti, "A review of anomaly detection in automated surviellance," *IEEE Trans. Syst., Man, Cybern.*, vol. 42, no. 6, pp. 1257–1272, Nov. 2012.

[22] Y. S. Chong and Y. H. Tay, "Modeling representation of videos for anomaly detection using deep learning: A review," 2015, *arXiv:1505.00523*.

[23] B. R. Kiran, D. M. Thomas, and R. Parakkal, "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos," *J. Imag.*, vol. 4, Jan. 2018, Art. no. 36.

[24] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1975–1981.

[25] B. Antic and B. Ommer, "Video parsing for abnormality detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2415–2422.

[26] H. Vu, T. D. Nguyen, T. Le, W. Luo, and D. Phung, "Robust anomaly detection in videos using multilevel representations," *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 5216–5223.

[27] B. Ramachandra, M. Jones, and R. Vatsavai, "Learning a distance function with a siamese network to localize anomalies in videos," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2587–2596.

[28] C. Lu, J. Shi, W. Wang, and J. Jia, "Fast abnormal event detection," *Int. J. Comput. Vis.*, vol. 127, no. 8, pp. 993–1011, 2019.

[29] S. Smeureanu, R. T. Ionescu, M. Popescu, and B. Alexe, "Deep appearance features for abnormal behavior detection in video," in *Proc. Int. Conf. Image Anal. Process.*, 2017, vol. 10485, pp. 779–789.

[30] R. T. Ionescu, S. Smeureanu, M. Popescu, and B. Alexe, "Detecting abnormal events in video using narrowed normality clusters," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 1951–1960.

[31] R. Herzig *et al.*, "Spatio-temporal action graph networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop*, 2019, pp. 2347–2356.

[32] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun, "Anticipating accidents in dashcam videos," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 136–153.

[33] Z. Che *et al.*, "D^2-city: A large-scale dashcam video dataset of diversetraffic scenarios," 2019, *arXiv: 1904.01975v2*.

[34] S. Haresh, S. Kumar, M. Zia, and Q.-H. Tran, "Towards anomaly detection in dashcam videos," in *Proc. IEEE Intell. Vehicles Symp.*, 2020, pp. 1–8.

[35] 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 Trans. Image Process.*, vol. 26, no. 4, pp. 1992–2004, Apr. 2017.

[36] J. M. Lobo, A. Jiménez-Valverde, and R. Real, "AUC: A misleading measure of the performance of predictive distribution models," *Global Ecol. Biogeography*, vol. 17, no. 2, pp. 145–151, 2008.

[37] V. Saligrama and Z. Chen, "Video anomaly detection based on local statistical aggregates," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2112–2119.

[38] K. Ma, M. Doescher, and C. Bodden, "Anomaly detection in crowded scenes using dense trajectories," 2015.

[39] Y. Cong, J. Yuan, and J. Liu, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognit.*, vol. 46, no. 7, pp. 1851–1864, Jul. 2013.

[40] K.-W. Cheng, Y.-T. Chen, and W.-H. Fang, "Video anomaly detection and localization using hierarchical feature representation and Gaussian process regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2909–2917.

[41] B. Antić and B. Ommer, "Spatio-temporal video parsing for abnormality detection," 2015, *arXiv:1502.06235*.

[42] S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2054–2060.

[43] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 935–942.

[44] R. Hinami, T. Mei, and S. Satoh, "Joint detection and recounting of abnormal events by learning deep generic knowledge," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3639–3647.

[45] 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 *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1689–1698.

[46] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked RNN framework," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 341–349.

[47] M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," *Comput. Vis. Image Understanding*, vol. 172, pp. 88–97, Jul. 2018.

[48] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe, "Learning deep representations of appearance and motion for anomalous event detection," in *Proc. Brit. Mach. Vis. Conf.*, 2015, pp. 8.1–8.12.

[49] M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, "Real-time anomaly detection and localization in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2015, pp. 56–62.

[50] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli, "Adversarially learned one-class classifier for novelty detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3379–3388.

[51] J. Kim and K. Grauman, "Observe locally, infer globally: A space-time MRF for detecting abnormal activities with incremental updates," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2921–2928.

[52] Y. Benezeth, P.-M. Jodoin, V. Saligrama, and C. Rosenberger, "Abnormal events detection based on spatio-temporal co-occurences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2458–2465.

[53] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1446–1453.

[54] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni, and N. Sebe, "Abnormal event detection in videos using generative adversarial nets," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 1577–1581.

[55] B. Ramachandra, "Anomaly detection in videos," Ph.D. thesis, Dept. of Comput. Sci., North Carolina State Univ., Raleigh, NC, 2019.

[56] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, and S. Venkatesh, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1705–1714.

[57] H. Park, J. Noh, and B. Ham, "Learning memory-guided normality for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14 372–14 381.

[58] R. Rodrigues, N. Bhargava, R. Velmurugan, and S. Chaudhuri, "Multi-timescale trajectory prediction for abnormal human activity detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2626–2634.

[59] A. Markovitz, G. Sharir, I. Friedman, L. Zelnik-Manor, and S. Avidan, "Graph embedded pose clustering for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 539–10 547.

[60] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional multi-person pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2334–2343.

[61] K. Cho *et al.*"Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2014, pp. 1724–1734.

[62] S. Zhou, W. Shen, D. Zeng, M. Fang, Y. Wei, and Z. Zhang, "Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes," *Signal Process.: Image Commun.*, vol. 47, pp. 358–368, Sep. 2016.

[63] Y. Zhu, "Motion-aware feature for improved video anomaly detection," in *Proc. Brit. Mach. Vis. Conf.*, 2019, Art. no. 12.

[64] J.-X. Zhong, N. Li, W. Kong, S. Liu, T. H. Li, and G. Li, "Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1237–1246.
[65] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–14.
[66] H. Tran and D. Hogg, "Anomaly detection using a convolutional winner-take-all autoencoder," in *Proc. Brit. Mach. Vis. Conf.*, 2017, Art. no. 139.
[67] Q. Sun, H. Liu, and T. Harada, "Online growing neural gas for anomaly detection in changing surveillance scenes," *Pattern Recognit.*, vol. 64, pp. 187–201, Apr. 2017.
[68] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 428–441.
[69] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.
[70] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.
[71] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *Int. J. Comput. Vis.*, vol. 103, no. 1, pp. 60–79, 2013.
[72] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. IEEE Int. Workshop Vis. Surveillance Perform. Eval. Tracking Surveillance*, 2005, pp. 65–72.
[73] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical Rev. E*, vol. 51, no. 5, 1995, Art. no. 4282.
[74] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Proc. 12th Int. Conf. Neural Inf. Process. Syst.*, 2000, pp. 582–588.
[75] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. Brit. Mach. Vis. Conf.*, 2014.
[76] A. Makhzani and B. J. Frey, "Winner-take-all autoencoders," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 2791–2799.
[77] F. Turchini, L. Seidenari, and A. Del Bimbo, "Convex polytope ensembles for spatio-temporal anomaly detection," in *Proc. Int. Conf. Image Anal. Process.*, 2017, vol. 10484, pp. 174–184.
[78] P. Casale, O. Pujol, and P. Radeva, "Approximate polytope ensemble for one-class classification," *Pattern Recognit.*, vol. 47, no. 2, pp. 854–864, 2014.
[79] B. Fritzke, "A growing neural gas network learns topologies," in *Proc. 7th Int. Conf. Neural Inf. Process. Syst.*, 1995, pp. 625–632.
[80] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[81] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2916–2929, Dec. 2013.
[82] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.
[83] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network," in *Proc. 6th Int. Conf. Neural Inf. Process. Syst.*, 1994, pp. 737–744.
[84] Y. Feng, Y. Yuan, and X. Lu, "Learning deep event models for crowd anomaly detection," *Neurocomputing*, vol. 219, pp. 548–556, Jan. 2017.
[85] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Comput.*, vol. 11, no. 2, pp. 443–482, 1999.
[86] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *Int. J. Comput. Vis.*, vol. 51, no. 2, pp. 91–109, 2003.
[87] A. B. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 909–926, May 2008.
[88] D. Gao and N. Vasconcelos, "Decision-theoretic saliency: Computational principles, biological plausibility, and implications for neurophysiology and psychophysics," *Neural Comput.*, vol. 21, no. 1, pp. 239–271, 2009.
[89] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn.*, 2001, pp. 282–289.
[90] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Berlin, Germany: Springer, 2003, pp. 63–71.
[91] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
[92] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[93] R. Krishna *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, 2017.
[94] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: A simple deep learning baseline for image classification?" *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5017–5032, Dec. 2015.
[95] A. Van den Oord and B. Schrauwen, "Factoring variations in natural images with deep gaussian mixture models," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 3518–3526.
[96] T.-N. Nguyen and J. Meunier, "Anomaly detection in video sequence with appearance-motion correspondence," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1273–1283.
[97] Y. Tang, L. Zhao, S. Zhang, C. Gong, G. Li, and J. Yang, "Integrating prediction and reconstruction for anomaly detection," *Pattern Recognit. Lett.*, vol. 129, pp. 123–130, Jan. 2020.
[98] S. H. I. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[99] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[100] H. Vu, D. Phung, T. D. Nguyen, A. Trevors, and S. Venkatesh, "Energy-based models for video anomaly detection," 2017, *arXiv: 1708.05211*.
[101] Y. Freund and D. Haussler, "Unsupervised learning of distributions on binary vectors using two layer networks," in *Proc. 4th Int. Conf. Neural Inf. Process. Syst.*, 1992, pp. 912–919.
[102] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.
[103] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.
[104] A. Dosovitskiy *et al.*, "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2758–2766.
[105] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1125–1134.
[106] Y. Zhang, H. Lu, L. Zhang, and X. Ruan, "Combining motion and appearance cues for anomaly detection," *Pattern Recognit.*, vol. 51, pp. 443–452, Mar. 2016.
[107] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 357–360.
[108] R. Leyva, V. Sanchez, and C.-T. Li, "Video anomaly detection with compact feature sets for online performance," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3463–3478, Jul. 2017.
[109] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6299–6308.

Bharathkumar Ramachandra (Member, IEEE) received the PhD degree from North Carolina State University, Raleigh, North Carolina, in 2020. He is currently a computer vision scientist at Wrnch AI in Montreal, Quebec. His research interests include machine learning and computer vision and in particular video anomaly detection, metric learning, deep generative modeling, and out-of-distribution generalization with neural networks.

Michael J. Jones (Senior Member, IEEE) received the PhD degree from the Massachusetts Institute of Technology, Cambridge, Massachusetts, in 1997. He was a member of the DEC's Cambridge Research Lab from 1997 to 2001 and is currently a senior principal research scientist at Mitsubishi Electric Research Laboratories (MERL), Cambridge, Massachusetts where he has been since 2001. His work has been awarded the Marr Prize and the Longuet-Higgins Prize. 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 reidentification, and video anomaly detection.

Ranga Raju Vatsavai (Member, IEEE) received the MS and PhD degrees in computer science from the University of Minnesota, Minneapolis, Minnesota. He is currently a chancellor's faculty excellence program cluster associate professor of geospatial analytics at the Department of Computer Science, North Carolina State University (NCSU), Raleigh, North Carolina, and a distinguished research fellow at Behavioral Reinforcement Learning Lab, Lirio. Before joining NCSU, he 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 national security, geospatial intelligence, natural resources, agriculture, climate change, location-based services, and human terrain mapping. He has published more than 100 papers in top ranking conferences and journals, and served on technical program committees of leading international conferences like KDD, ICDM, SDM, ACM SIGSPATIAL GIS, AAAI, WACV, and ECML/PKDD.