Person Fall Detection Using Weakly Supervised Methods

Kjartan Madsen [1,2]
kjartan.madsen9@gmail.com

Zenjie Li[2]
zli@milestone.dk

Francois Lauze[1]
francois@di.ku.dk

Kamal Nasrollahi[2,3]
kn@create.aau.dk

University of Copenhagen[1] Milestone Systems[2] Aalborg Universitet[3]

Abstract

Person falls can result in severe injuries or fatalities. An automatic fall detection system can potentially save lives by promptly alerting others. Existing fall detection methods that employ physical sensors like accelerometers have limitations. Current computer vision-based approaches, trained on simple and unrealistic datasets, also lack effectiveness. Creating a new dataset for traditional supervised learning would require a significant amount of time to annotate. To address this, we adopt weakly supervised methods from Video Anomaly Detection (VAD) and curate a high-quality and realistic dataset. Our proposed model, utilizing Multiple Instance Learning, introduces a novel loss function that outperforms state-of-the-art anomaly detection models for fall detection. Furthermore, despite its simplicity, our approach achieves competitive performance compared to the current state of the art in UCF-Crime.

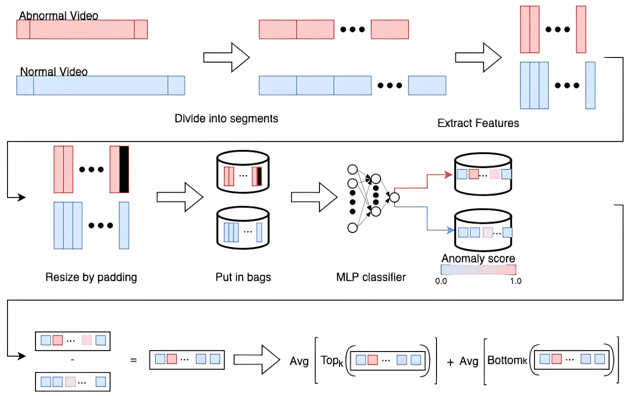


Figure 1. The Multiple Instance Learning illustrated. Features are extracted from each video and each feature set (one set per video) is made into the same length. Finally, one feature set is sampled from each class and used as input to a classifier network that is trained using our proposed loss function.

1. Introduction

A person falling could lead to severe injury and even fatalities. Falling was the second leading cause of unintentional injury deaths worldwide in 2020 [20]. Reports from the USA [20] and EU [19] found that the older demographic is especially prone to suffer serious consequences from falling.

The development and adoption of a reliable fall detection system would have big financial, safety, and health benefits. A reliable system reduces the amount of time staff needs to check in on patients, making time for other duties or even reducing the size of the staff needed. Such a system would also result in faster response to fallen and potentially injured patients which could help to reduce the severity of injuries sustained. A brief online search reveals that the predominant methods for automatic fall detection in the healthcare industry heavily rely on accelerometers[1][2][3] and comparable devices.

These devices could lead to discomfort and require individual fitting. They might necessitate recharging and have a limited range. In contrast, camera-based detection, aided by existing hospital security cameras, offers easy deployment without patient fitting or maintenance. Video systems allow swift remote alarm verification and can function universally across varied camera-equipped settings.

Anomaly Detection (AD) in videos or images involves identifying events or patterns that deviate from the norm. The definition of an *anomaly* is subjective and context-dependent. AD offers an advantage by not necessitating examples of all potential anomalies, unlike methods like classification or action recognition, which demand extensive annotated data per class/action. Moreover, the AD approach detects undefined classes, encompassing situations that require alarms, including circumstances posing challenges in data collection.

One popular approach in VAD is unsupervised anomaly detection. Here, the model exclusively learns from normal data, detecting anomalies via higher reconstruction er-

[1] www.medicalguardian.com
[2] www.lifeline.com/medical-alert-systems/fall-detection/
[3] www.ncoa.org

rors for unseen cases. Typically, these unsupervised approaches are scene-dependent, trained and tested within a single scene. Our goal is to create a scene-agnostic model, applicable across diverse locations. While some scene-agnostic, unsupervised AD models exist [7], they often rely on multiple complex feature extractors, making deployment impractical.

In this study, we investigate weakly supervised anomaly detection, leveraging partial anomaly information during training. We propose a cost-effective approach for detecting person falls, distinct from conventional supervised techniques like image classification or action recognition, which demand extensive annotations. Our method significantly reduces data collection expenses. We employ the Multiple Instance Learning (MIL) framework and introduce a novel loss function utilizing rank statistics (top_k and $bottom_k$) to distinguish normal and abnormal features in weakly labeled videos. Our proposed loss function enhances the results of the Multiple Instance Learning setup from [16] in both the newly collected fall dataset and the widely-used UCF-Crime anomaly detection dataset [16]. The top_k function has been previously employed in Multiple Instance Learning, notably in [17] and [12]. In [17], the Video Anomaly Detection (VAD) model leverages the difference between the top_k abnormal snippets and top_k hardest normal snippets. The "Soft bag MIL" proposed in [12] employs a probabilistic approach to define "soft bags" and trains an SVM classifier to extract the top k instances in an image classification task. In contrast, we calculate the disparity between the two bags and utilize statistical rank functions to distinguish k instances within the abnormal bag from the rest (see Sec. 4 for details).

Our contributions include:

- The Our+VFPK+UR dataset, a high-quality and realistic dataset for fall detection and anomaly detection algorithms.
- A novel loss function for MIL in our model using zero-padded TimeSformer features, outperforming SOTA VAD models on fall detection and achieving competitiveness on UCF-Crime.
- Demonstrating the efficacy of fixed-length zero-padding for improved performance in Multiple Instance Learning with a proper feature extractor.

The rest of the paper is organized as follows. Section 2 outlines relevant prior research on Fall Detection and VAD. Section 3 introduces our new dataset. Section 4 presents Multiple Instance Learning and our novel loss function. Section 5 elaborates on experiments and results. In Section 6, we discuss results and future prospects. Finally, section 7 concludes our research.

2. Related work

This section presents a review of prior research in vision-based fall detection and the domain of Video Anomaly Detection.

2.1. Fall Detection

An *et al.* [3] compare multiple classifiers trained on the two-class problem (fall and non-fall) and compare it to the one-class approach where they trained only on frames displaying falls. They test various classification networks for both the two-class and one-class problems and find that the models trained on both classes are slightly better at detecting falls. Adhikari *et al.* [1] apply a 3-Dimensional CNN to fall detection. They extract a human silhouette from the raw RGB images, and use the CNN to detect its different poses. They classify a fall as a sequence of poses that end in a "lying" pose. Alanazi and Muhammad [2] explicitly incorporate temporal information in a multi-stream 3D CNN architecture. Their model takes a stack of temporally fused frames in each stream. They achieve very competitive performance compared to other state-of-the-art fall detection models and fine-tuned classification networks.

These approaches demand a substantial amount of labeled data, yet the datasets on which they are trained and evaluated do not encompass scenes complex enough to thoroughly assess the real-world performance of each model. In contrast, our method has the advantage of being capable of learning from weakly labeled data. That makes the collection of a new, high-quality dataset for the development of a dependable real-world fall detection system a considerably more economical process.

2.2. Anomaly Detection

Sultani *et al.* [16] frame anomaly detection as a Multiple Instance Learning problem (MIL) (more on MIL in Section 4). MIL relaxes the assumption of one label for each instance of data, which means that we can train on weakly labeled data. MIL in the case of anomaly detection in videos means that one labels each video as either containing an anomaly (positive) or not (negative), so the exact instances that are anomalous are unknown. With this approach, the authors propose a simple architecture consisting of a 3D CNN feature extractor, and a fully connected neural network trained on a special MIL ranking loss function of their design. They show that their approach significantly outperforms earlier anomaly detection algorithms on their proposed dataset, UCF-Crime.

The MIL loss function proposed by Sultani *et al.* [16] is:

$$l(\mathbf{B}_A, \mathbf{B}_N) = l_1 + l_2 + l_3 \tag{1}$$

$$l_1(\mathbf{B}_A, \mathbf{B}_N) = \max(0, 1 - \max_i f(V_A^i) + \max_i f(V_N^i)) \tag{2}$$

$$l_2(\mathbf{B}_A, \mathbf{B}_N) = \lambda_1 \sum_{i=1}^{n-1} (f(V_A^i) - f(V_A^{i+1}))^2 \quad (3)$$

$$l_3(\mathbf{B}_A, \mathbf{B}_N) = \lambda_2 \sum_{i=1}^{n} f(V_A^i) \quad (4)$$

where $f(\cdot)$ is the MLP and V_A^i denotes a clip with index i in $\mathbf{B}_A$. Similarly, V_N^i denotes the clip with index i in $\mathbf{B}_N$. λ_1 and λ_2 are hyperparameters. The first term in $l(\mathbf{B}_A, \mathbf{B}_N)$, called the *Hinge loss*, aims to maximize the activation of one of the clips in $\mathbf{B}_A$, presumably the clip containing the anomalous event, and concurrently minimize the highest activation in $\mathbf{B}_N$. The second term is a temporal smoothness term, trying to enforce smooth activation of sequential video clips in $\mathbf{B}_A$. The last term is a sparsity term, capturing the assumption that the majority of clips in the abnormal video are actually normal. Our approach extends this method by incorporating the *top-k* and *bottom-k* rank functions, to capture the fact that an anomaly might not be contained in a single clip.

The authors of [17] contend that MIL brings forth four issues: 1) the highest anomaly score might not correspond to an abnormal snippet, 2) simple fitting of normal data can hinder training convergence, 3) videos with multiple abnormalities are underutilized, 4) employing a classification score yields a weak training signal. To tackle these, they devise a loss function that enhances the distinction between normal and abnormal videos, by maximizing the separability of magnitudes of the *top-k* snippet features. They name this *feature magnitude learning*. This, together with a novel method for extracting global temporal features using *dilated convolution*, compromises their method *Robust Temporal Feature Magnitude learning (RFTM)*. RFTM reaches state-of-the-art on a number of VAD datasets, including UCF-Crime [16]. RFTM separates the magnitude of the features, whereas our model is trained to separate normal and anomalous clips based on the final anomaly score. Furthermore, their approach uses a classifier with dilated convolutions, which is more complicated and resource-hungry than our simple neural network.

Wu *et al*. [21] propose a model they call *self-supervised sparse representation* (S3R for short) that reached state-of-the-art in UCF-Crime [16] and XD-Violence [22]. S3R unifies dictionary learning and a reconstruction-based approach. S3R learns a normal-event dictionary and utilizes two modules: *en-Normal* that aims to reconstruct the dictionary features, and *de-Normal* that learns to filter out the normal features from the snippets. Lv *et al*. [13] propose the Weakly Supervised Anomaly Localization (WSAL) model for precise anomaly localization in time series. WSAL estimates anomaly localization to inform a MIL margin function, utilizing feature differences between neighboring instances. This expects abnormal videos exhibiting larger maximal differences than normal ones. Additionally, they introduce High-Order Context Encoding to capture semantic features and encode temporal variations in videos.

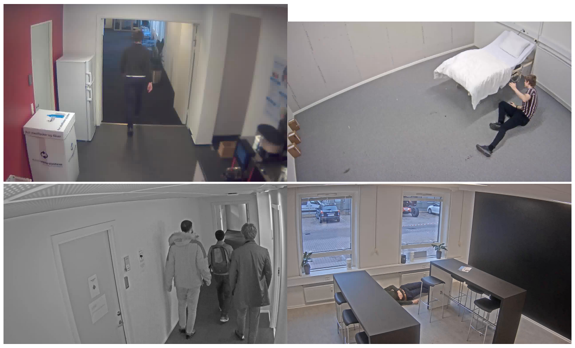


Figure 2. Examples from the data collected by us.

Although S3R and WSAL exhibit strong VAD performance, their complex architectures might hinder inference speed and elevate hardware prerequisites, thereby increasing costs. Our approach proves comparable or superior in fall detection while maintaining a simpler design.

3. Dataset

A few vision-based public datasets for fall detection are accessible online. However, these typically offer limited scenes and subjects. Some Human Action Recognition (HAR) datasets like [8], [10], and [14] include the “falling” class. Yet, these often comprise YouTube or movie clips, posing issues such as abrupt angle changes, multiple actions, camera shake, and non-human subjects [14]. Additionally, each video is trimmed to showcase only a single action, rendering them unsuitable for anomaly detection. To the best of our knowledge, no HAR dataset captures falls through standard surveillance camera setups.

3.1. Our Our+VFPK+UR Dataset

As there are no suitable existing datasets for building a functional fall detection model using the Anomaly Detection framework, we’ve gathered a new dataset designed for fall and anomaly detection. This compilation includes normal surveillance videos and abnormal surveillance videos containing falls, along with selected samples from VFPK [3] and UR-Fall dataset [11].

The **Vision-based fallen person dataset (VFPK)** [3] contains high-resolution videos emulating real CCTV clips of people falling, or lying down, in public areas. It features varied viewpoints, backgrounds, and lighting, yet many videos have pronounced camera shake, or show individuals lying down rather than falling. By filtering out videos with such issues, we’ve curated a small, relevant subset.

The **UR-Fall detection dataset** [11] dataset combines depth and RGB images for fall detection. Its simplicity, with limited scene variety and individuals, enabled authors to attain nearly 100% accuracy using a basic SVM classifier trained on RGB + depth data. Notably, the dataset features high-quality fall scenarios and camera setups, curated within the Our+VFPK+UR dataset.

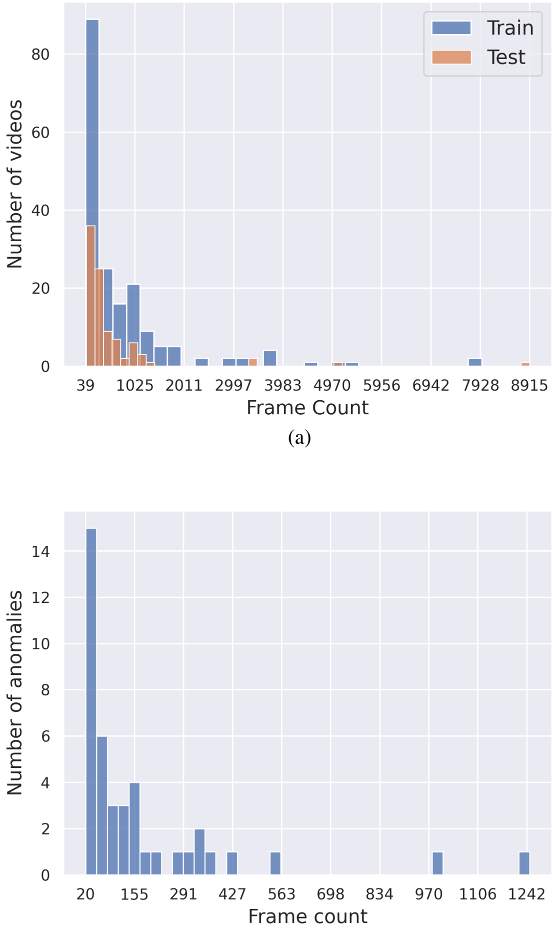
Our dataset consists of high-resolution videos (1080p) captured indoors across various locations: a mock hospital room, a real hospital room, an office building's waiting room, and five corridors. Each location has a stationary surveillance camera, except for the mock hospital room, which has five cameras. The normal videos depict people engaged in everyday activities such as walking, standing, sitting, talking, and lying in bed. Atypical postures like squatting and kneeling are also included. Abnormal videos portray individuals falling in different scenarios, including rolling off a bed or couch, falling from standing or sitting positions, and rolling on the floor. The falls are evident, lifelike, and prominently displayed. The dataset covers both normal and low-light conditions. A total of 19 characters are featured across the scenes, and all participants provided consent for their involvement. Refer to Figure 2 for visual examples from this collection.

Dataset	Normal Videos	Abnormal Videos
VFPK	0	6
UR-Fall	46	52
Our	97	46
Total	**143**	**104**

Table 1. Videos collected from the different sources used in the Our+VFPK+UR dataset.

The resulting dataset comprises 143 normal videos and 104 anomaly videos. Refer to Table 1 for the video distribution across UR-Fall, VFPK, and our sources. All videos are sampled at 30 frames per second. Each class's videos were randomly divided into training and test sets, aiming for a balanced 75-25 split (147,230 frames for training, 54,560 for testing). Anomaly videos typically include one or two anomalous events of varying durations. Figure 3 illustrates the distribution of video lengths in the train/test sets and anomaly lengths in the test set.

The dataset aligns with UCF-Crime's format [16], allowing weakly supervised Video Anomaly Detection (VAD) using video-level labels for training and frame-level annotations for testing. Fall annotations are applied as soon as falls become evident, ensuring rapid response for fall detection systems. Annotations were carried out by a single annotator. The anomaly event extends until the fallen person stands up. To encompass the initial stages of fall sequences (crucial for early detection), our annotator re-annotated samples from the VFPK and UR-Fall Detection datasets, which were initially annotated for frame-based methods.



(a)

(b)

Figure 3. The distribution of (a) the video lengths in the train/test sets and (b) the anomaly lengths in the test set in the Our+UR+VFPK dataset. Most fall events (anomalies) last a few dozen frames, equating to 1-2 seconds at a video frame rate of 30 frames per second.

4. Method

We adopt the approach by Sultani *et al.* [16], known as MIL (Multiple Instance Learning). Each video is divided into N segments/clips of fixed length. The segments from both videos are then placed in two separate "bags". $\mathbf{B}_A$ denotes the bag containing the clips from the abnormal video, and $\mathbf{B}_N$ denotes the bag containing the clips from the normal video. Each bag contains N clips. A pretrained feature extractor is used to extract features from the clips in the bags. Specifically, we utilize a pre-trained action recognition feature extractor. The rationale behind this choice is that the feature extractor is trained to extract features crucial for human action classification, which is assumed to be important in learning to detect people falling in VAD settings. Moreover, HAR feature extractors are trained using uncropped frames, eliminating the need for an object detector.

During training, a mini-batch comprises a random selection of $2M$ videos from the training data, including M normal and M abnormal videos. The feature extractor derives a feature set from each video. After resizing this set to a fixed size of N, it is placed in either $\mathbf{B}_A$ or $\mathbf{B}_N$, based on the video's class. For videos longer than N, features are averaged using a non-overlapping window, determined by the feature set and N, ensuring aligned consecutive features. For videos shorter than N, features are duplicated to reach the length N. The MIL approach is depicted in figure 1. To prevent feature compression or repetition, our study also applies zero padding (see Sec. 5.2 for specifics).

Sultani et al.'s original method exhibits limitations. Their hinge loss (Eq. (2)), employing the $max(\cdot)$ function, assumes a single-instance anomaly, which may not be accurate. To address this, we advocate employing the statistical rank functions top_k and $bottom_k$ to enable the model to learn that anomalies could span multiple instances. These functions are implemented into PyTorch in a differentiable way [15]; during the forward pass the indices of the k largest/smallest (top_k/$bottom_k$) are stored, and during the backward pass the gradient is exclusively computed for these indices only, effectively simplifying the operation to a vector-matrix product.

We optimize by distinguishing between normal and abnormal videos. In training, we sample one video from each class with uniform probability. Over numerous iterations, each normal video is effectively paired with each abnormal video, enabling the model to learn the distinctions between them.

Our proposed loss function is then:

$$l_{diff}(\mathbf{B}_A, \mathbf{B}_N) = l_{bot} + l_{top} \tag{5}$$

$$l_{bot} = mean(bottom_{k_1}[f(V_N) - f(V_A)]) \tag{6}$$

$$l_{top} = mean(top_{k_2}[f(V_N) - f(V_A)]) \tag{7}$$

We call this the *Difference-loss*. The $bottom_{k_1}$ and top_{k_2} are vectors of size k_1 and k_2 respectively, and $mean(\cdot)$ is the arithmetic mean of the elements of the argument. l_{bot} and l_{top} are in the range of [-1, 1] ($f(V_{A/N}) \in [0, 1]$ due to a Softmax on the output of the last layer in the MLP). The *Difference-loss* aims to separate the abnormal clips in the abnormal videos from the rest of (the normal part of) the abnormal video, as well as from the normal video. Intuitively, assuming the anomaly is contained in k_1 instances in $\mathbf{B}_A$, the term l_{bot} ensures that k_1 instances in $\mathbf{B}_A$ stand out from the instances in $\mathbf{B}_N$. l_{top} leads to k_2 instances in $\mathbf{B}_A$ and $\mathbf{B}_N$ get close to each other, i.e., close to zero. l_{bot} and l_{top} serve to guarantee the inclusion of both normal clips in $\mathbf{B}_N$, normal clips in $\mathbf{B}_A$, as well as the anomalous clips in $\mathbf{B}_A$ during the training process. The complete loss function inherits l_2 and l_3 from equation 1:

$$l'(\mathbf{B}_A, \mathbf{B}_N) = \max(0, \omega + l_{diff}) + l_2 + l_3 \tag{8}$$

$$\text{Top-k-Difference-Loss} = l'(\mathbf{B}_A, \mathbf{B}_N) + \|\boldsymbol{w}_f\| \tag{9}$$

We use $max(0, \cdot)$ together with a hyperparameter $\omega \in [0, 1]$ to control the maximum contribution of l_{diff}; a larger ω means l_{diff} is contributing more to the final loss function (Eq. (9)).

Taking the extremum (max/min) over the difference $f(V_N) - f(V_A)$ is not the same as taking the difference over the extremum since the max/min (and similarly the top_k and $bottom_k$) functions are not commutative. The original approach in [16] aims to maximize the response of the network to a single segment in each of the anomalous videos and minimize the response of the network to anything in the normal videos. The intuition behind equation 8 is to maximize the difference of segments in $\mathbf{B}_A$ from both the rest of the anomalous video and the normal video.

The classifier we employ is identical to the one in [16]. It comprises a three-layer MLP with *Dropout*. In their study, [16] explored various depths and sizes, determining that the three-layer deep MLP yielded optimal results.

5. Experiments

5.1. Zero-Shot HAR Baseline

The TimeSformer [4] with divided space-time attention pre-trained on Kinetics-600 [5] is used as a baseline for our dataset. The TimeSformer achieves an accuracy of 81.8% on the Kinetics-600 action recognition dataset. The Kinetics-600 dataset has two classes describing a fall event: "falling off bike" and "falling off chair". This means that if the TimeSformer predicts either of these classes, then it is correct. The model will not be further trained and will only be tested on the test set. It is expected to provide a weak baseline for our new dataset.

5.2. Feature Extraction Study

We conduct a study to determine the best-performing feature extractor on our dataset. The features are extracted in sets of 16 consecutive frames (i.e., the step length is 1). The feature set from each video is then resized to fit into a bag of size 32, as described in Section 4. The model is trained on our dataset using the proposed method. The results are listed in Table 2.

The experiment shows that the 3dResNet features perform the best, with a slight advantage over the TimeSformer

Model	dataset	AUC	Feature size
C3D [18]	Kinetics-400	0.84	4096
TimeSformer [4]	Kinetics-400	0.84	768
TimeSformer [4]	Kinetics-600	0.86	768
3dResNet-152 [9]	Kinetics-400	**0.87**	2048

Table 2. Results of using different feature extractors with features of fixed size (=32) on the Our+VFPK+UR Dataset. The dataset is the one it is pre-trained on.

Bag size	Feature Extractor	AUC
20	3dResNet	0.87
32	3dResNet	0.87
50	3dResNet	0.87
zero-pad	3dResNet	0.85
zero-pad	TimeSformer	0.90

Table 3. The results of using different segment lengths and zero-padding. Results obtained using Our+VFPK+UR Dataset.

that is pre-trained on Kinetics-600, while the C3D and the TimeSformer pre-trained on the Kinetics-400 perform the worst. The remaining results are obtained with the 3dResNet features unless otherwise stated.

We further investigate how the performance varies with different bag sizes. Many works only ever explore one bag size. We examine the impact of varying bag sizes and the use of zero padding to a fixed length on the performance of the MLP classifier. Zero padding is motivated by the use of padding in language transformer architecture to make the input sequence the correct size. Without zero padding, longer videos have their features averaged during bagging, leading to a lower resolution. Conversely, shorter videos have their features repeated. It is noted that zero padding is both in the normal and abnormal bags, so the model is expected to learn the relevant clips as normal (either in a normal or abnormal video). The results are listed in Table 3. Table 3 shows that the bagging of the features does not play a big role in the performance of the classifier. Surprisingly, the size of the bags does not seem to matter at all, and we found the bagging approach to be better than zero padding for the 3dResNet features. We also found that with the TimeSformer, the zero padding is better than both the bagged 3dResNet and the bagged TimeSformer features (see Table 2). Following the other works [16] [21], we stick to a bag size of 32 for the 3dResNet features.

5.3. Fall Detection

This section contains the main experiment where we explore multiple VAD methods as fall detection systems using the Our+VFPK+UR dataset presented in Section 3. The hyperparameters for WSAL [13] and S3R [21] are the same as those used to achieve the reported performance on UCF-Crime [16] in their respective original works.

Method	AUC %	AP %
HAR baseline	51.13	34.86
WSAL [13]	84.29	31.26
S3R [21]	88.21	52.38
MIL [16]	83.87	40.83
Ours	87.24	45.86
Ours+zpTf	**89.76**	**54.09**

Table 4. Results on the Our+VFPK+UR dataset. The best result is highlighted by bold numbers. Blue and red highlight the second and third best, respectively.

The results listed in Table 4 show that the HAR baseline is indeed weak, performing only slightly better than random. WSAL and the original MIL approach by [16] achieve a score of 84.29% and 83.87% AUC, respectively. S3R is the second best model in our testing with an AUC reaching 88.21% AUC and scoring 52.38% in AP on our dataset. Our model, with $k_1 = 3$, $k_2 = 2$, $\lambda_1 = \lambda_2 = 10^{-4}$, $\omega = 0.1$, and dropout rate $= 0.6$, performs the best when using the zero-padded TimeSformer, in terms of both AUC (89.76%) and AP (54.09%). When using the 3dResNet features, our model ranks third in both AUC (87.24%) and AP (45.86%).

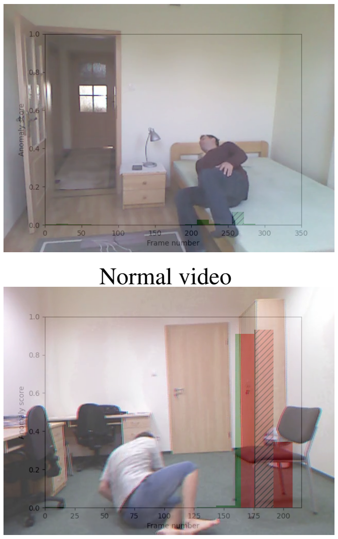
We conduct ablation studies to evaluate the effectiveness of our novel loss functions, as outlined in equation 5. Keeping all other hyperparameters unchanged, the utilization of only l_{bot} results in decreased AUC (68.37%) and AP (32.06%), while using solely l_{top} leads to reduced AUC (86.47%) and AP (38.71%).

Figure 4 shows two qualitative examples of anomaly detection using our model with TimeSformer feature extraction and zero padding.

5.4. Model Fit Study

We conduct experiments where each model is trained and tested on the test set. This means that the model is trained using weak labels and subsequently tested on the same dataset. This is to investigate whether the weak labels are enough for each model to learn the correct labels. How well each model is able to fit the true labels only using the "weak"/video-level labels gives an indication of how strong the training signal in each model is.

All models reach a better AUC score in this experiment. The gains in AUC are modest for WSAL (1.09%), MIL (0.25%), and S3R (0.14%), whereas our method shows a more significant performance gain with 3dResNet features (2.48%), but only a small gain (0.54%) using the zero-padded TimeSformer features. Furthermore, MIL, S3R and our model with the zero-padded TimeSformer features all



Normal video

Person fall

Figure 4. Anomaly detection in the UR-Fall dataset using our model with TimeSFormer feature extraction and zero padding. Each image shows an example: the starting frame of a 16-frame clip (crosshatch bar). The histogram displays anomaly scores for all clips in the video, with green bars indicating normal clips and red bars indicating anomalies.

Method	AUC %	AP %
S3R	88.79 (88.21)	52.16 (52.38)
Ours	87.36 (87.24)	43.87 (45.86)
Ours+zpTf	**89.39** (89.76)	**55.00** (54.09)

Table 5. Results of adding more normal data to training. Values in parentheses are the scores obtained with the original training set, i.e. without additional training data. The best result is highlighted by bold numbers and blue highlights the second best.

perform worse in AP, whilst our model with the 3dResNet features gains 7.09 percentage points, and WSAL gains 4.90 percentage points. None of the models are able to overfit.

5.5. Performance Scaling With More Normal Videos

Anomaly detection algorithm effectiveness should ideally scale, to a certain extent, with the quantity of normal training data. Particularly, an increase in normal data should lead to a reduction in the false positive rate. Consequently, we supplement the training dataset with 30 additional Normal Videos (33,910 frames) and evaluate the model's performance with and without this supplementary data. The resultant dataset is approximately 22.3% larger than the original. Table 5 illustrates the comparative performance of S3R [21], our method, and their variations involving the extra normal data.

Method	AUC %	AP %
S3R [21]	84.05	6.89
MIL [16]	74.56	4.40
Our	81.62	7.69
Our+zpTf	**90.78**	**38.95**

Table 6. The outcomes of employing VAD models for fall action detection. The best result is indicated in bold, with the second and third best results highlighted in blue and red, respectively.

The results of using more normal data for training are mixed. S3R gains 0.58% in AUC, but loses 0.22% in AP. Our model with the 3dResNet features gains an insignificant 0.12% in AUC and loses 1.99% in AP. Our model with the zero-padded TimeSformer features shows a small decrease in AUC (0.37%) and an increase in AP (0.91%). This shows that these methods are not able to properly utilize "free" normal data. We hypothesize that their learning is limited by the increased class imbalance in the dataset.

Increased normal data yield mixed training outcomes. S3R sees a 0.58% AUC improvement but a 0.22% AP decrease. Our model with 3dResNet features gains a mere 0.12% AUC while losing 1.99% AP. Our model with zero-padded TimeSformer features exhibits a slight AUC dip of 0.37% and an AP increase of 0.91%. Evidently, these methods struggle to leverage the surplus normal data effectively, likely due to restricted learning imposed by heightened class imbalance.

5.6. Falling Action Detection

Alternatively, a fall detection system can exclusively target fall actions, excluding the identification of a person on the ground. We revise the test set annotations, shortening anomaly detection to 32 frames (approximately 1 second) after the fall begins. Table 6 displays the outcomes of this event-focused method. All results are derived with the same hyperparameters fine-tuning as in Section 5.3.

S3R [21], MIL [16], and our 3dResNet-based model all demonstrate notably low AP scores, underscoring their limited capacity to effectively capture the falling action. The substantial decline in AP values outlined in Table 4 implies that these models excel more in detecting the fallen person than the actual falling action. In contrast, our model leveraging zero-padded TimeSformer features achieves the highest performance in both metrics, demonstrating its superior capability in capturing the essence of a fall.

We also test the original MIL [16] and our model with TimeSformer [4] features (without zero padding) with a bag size of 32. MIL with the TimeSformer features is able to achieve 84.63% AUC and 28.11% AP and our model got 83.28% AUC and 23.20% AP. This highlights the significant role of the feature extractor in detecting the falling ac-

Model	FE	AUC %
MIL [16]	C3D	75.41
MIL [16]	3dResNet	82.00
MGFN [6]	I3D	**86.98**
S3R [21]	I3D	85.99
WSAL [13]	TSN	85.38
RFTM [17]	I3D	84.30
Ours	3dResNet	84.71

Table 7. Results on UCF-crime. The best result is highlighted by bold numbers while blue and red highlight the second and third best results, respectively.

tion. Additionally, our model's performance benefits from zero padding the features, as demonstrated by these results.

5.7. Performance on UCF-Crime

We evaluate the generality of our approach as an anomaly detector using the renowned **UCF-Crime** dataset [16]. This dataset features extensive untrimmed surveillance videos encompassing 13 real-world scenarios, like shoplifting, fighting, accidents, and arrests, all labeled as anomalies. It comprises 1900 annotated videos from the internet, half of which involve anomalies. Training videos have video-level annotations while testing videos have frame-level annotations.

We also conduct experiments on UCF-Crime [16], to gauge the generality of the proposed method. We use the 3dResNet features with a fixed size of 32 segments to obtain our results. Hyperparameters are $k_1 = k_2 = 1$, $\omega = 0.1$, $\lambda_1 = 2 \cdot 10^{-4}, \lambda_2 = 2 \cdot 10^{-3}$. During our model testing, we reset the remaining frames to zero after division by 16, following the dataset authors' approach [16]. The results of our approach, along with the state of the art, are presented in Table 7. Our approach ranks among the top 4 according to paperswithcode.com[4].

Utilizing our identical model as presented in Table 7, we assess it on the Our+VFPK+UR Dataset, yielding a modest 67.37% AUC and 32.63% AP. This underscores the critical need for a new, superior-quality fall detection dataset.

6. Dicussion

In this paper, we collect a novel dataset comprising both normal videos and those depicting a person falling. The dataset is labeled with video-level annotations. Extending the Multiple Instance Learning approach by [16], we introduce an innovative loss function utilizing statistical rank functions, namely top_k and $bottom_k$, for effective differentiation between normal and abnormal video segments. Our model, employing the zero-padded TimeSformer [4] features, outperforms alternative models in ROC AUC and AP metrics on our proposed fall detection dataset. Furthermore, our approach exhibits superior capability in detecting the initiation of falling events compared to other techniques. Notably, our method achieves a top-4 ranking in the widely-used UCF-Crime anomaly detection dataset [16], underscoring its effectiveness, despite its simplicity.

Furthermore, it is noteworthy that in situations like common anomaly detection setups with notable class imbalance, Average Precision should be prioritized over ROC AUC for assessing model performance. This is due to the abundance of negative samples. While a model with consistently high false positives and low false negatives might exhibit a high ROC AUC, its precision is likely to be low.

We highlight key findings from our study. Firstly, we enhance the basic MIL approach to fall detection, surpassing Sultani *et al.*'s [16] model. By refining the feature extractor and loss function, we elevate the AUC from 75.41% (UCF-Crime [16]) to 84.72%. This uncomplicated method has the potential for further performance improvements through the exploration of supplementary features, classifiers, and variations of the suggested loss function detailed in this paper. Secondly, we identify a vulnerability in the existing state-of-the-art VAD models. Specifically, they fail to demonstrate enhancement when introduced to supplementary normal data. Our proposed variant and the S3R model [21] show no advancement in terms of AP and AUC even with the inclusion of an extra $\approx 22\%$ of normal data.

Future research can extend our study by investigating diverse classifier architectures, assessing our model's ability to detect novel anomalies, and utilizing annotations from datasets like UR-Fall [11] for training with weak labels. To reduce false positive rates and enhance recall, an alternative strategy could involve an ensemble that combines our solution with a frame-based classification model trained on the fall datasets referenced in this paper.

7. Conclusion

VAD techniques based on Multiple Instance Learning and weakly supervised learning showed great promise in accurate fall detection. Feature extractors and the loss functions play a great role in the performance. However, striking a balance between minimizing false alarms and effectively detecting most falls remains a challenge for SOTA models. Continued efforts are needed to enhance the performance of these systems.

References

[1] Kripesh Adhikari, Hamid Bouchachia, and Hammadi Nait-Charif. Activity recognition for indoor fall detection using convolutional neural network. pages 81–84, 05 2017. 2

[4] https://paperswithcode.com/sota/anomaly-detection-in-surveillance-videos-on

[2] Thamer Alanazi and Ghulam Muhammad. Human fall detection using 3d multi-stream convolutional neural networks with fusion. *Diagnostics*, 12:3060, 12 2022. 2

[3] Jaeju An, Jeongho Kim, Hanbeen Lee, Jinbeom Kim, Junhyung Kang, Minha Kim, Saebyeol Shin, Minha Kim, Donghee Hong, and Simon S. Woo. VFP290k: A large-scale benchmark dataset for vision-based fallen person detection. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 2, 3

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021. 5, 6, 7, 8

[5] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600, 2018. 5

[6] Yingxian Chen, Zhengzhe Liu, Baoheng Zhang, Wilton Fok, Xiaojuan Qi, and Yik-Chung Wu. Mgfn: Magnitude-contrastive glance-and-focus network for weakly-supervised video anomaly detection, 2022. 8

[7] Mariana Iuliana Georgescu, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. A background-agnostic framework with adversarial training for abnormal event detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):4505–4523, 2022. 2

[8] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. Ava: A video dataset of spatio-temporally localized atomic visual actions, 2017. 3

[9] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition, 2017. 6

[10] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011. 3

[11] Bogdan Kwolek and Michal Kepski. Human fall detection on embedded platform using depth maps and wireless accelerometer. *Computer Methods and Programs in Biomedicine*, 117(3):489–501, 2014. 3, 8

[12] Wei-Xin LI and Nuno Vasconcelos. Multiple instance learning for soft bags via top instances. 06 2015. 2

[13] Hui Lv, Chuanwei Zhou, Zhen Cui, Chunyan Xu, Yong Li, and Jian Yang. Localizing anomalies from weakly-labeled videos. 2021. 3, 6, 8

[14] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, and Aude Oliva. Moments in time dataset: one million videos for event understanding, 2018. 3

[15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 12 2019. 5

[16] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos, 2018. 2, 3, 4, 5, 6, 7, 8

[17] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W. Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning, 2021. 2, 3, 8

[18] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 6

[19] S. Turner, R. Kisser, and W. Rogmans. Falls among older adults in the eu-28: Key facts from the available statistics, 2015. 1

[20] WHO. Worlds health organization, 2021. 1

[21] Jhih-Ciang Wu, He-Yen Hsieh, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Self-supervised sparse representation for video anomaly detection. In *ECCV*, 2022. 3, 6, 7, 8

[22] Peng Wu, jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only look, but also listen: Learning multimodal violence detection under weak supervision. In *European Conference on Computer Vision (ECCV)*, 2020. 3