Zero-shot Video Change Detection for Real-life Industrial Applications

Mahbubul Alam, Huimin Zhuge, Teresa Gonzalez, Ahmed Farahat, Song Wang, and Chetan Gupta

Industrial AI Lab, Research & Development, Hitachi America Ltd., Santa Clara, CA, 95054, USA
mahbubul.alam@hal.hitachi.com
joy.zhuge@hal.hitachi.com
teresa.gonzalezdiaz@hal.hitachi.com
ahmed.farahat@hal.hitachi.com
song.wang@hal.hitachi.com
chetan.gupta@hal.hitachi.com

Abstract

Change detection is crucial for various industrial applications. Although image change detection datasets are abundant, the collection of labeled video data is time-consuming, expensive, and cumbersome. This scarcity of labeled data motivates the development of few-shot or zero-shot video change detection techniques which may generalize well to new situations. Existing video change detection methods require large amounts of labeled data, are task-specific, and difficult to generalize. Therefore, in this paper, we propose a zero-shot video change detection algorithm using pre-trained deep learning models and conventional image processing techniques. Our approach identifies matching frames from input videos, adjusts lighting conditions if necessary, and uses an existing object detection model to identify objects in both frames. The method is easily generalizable by making few changes. We evaluate our proposed method on the VDAO dataset collected in a cluttered industrial environment and demonstrate its effectiveness in detecting changes between pairs of videos containing single and multiple objects.

1. Introduction

Video change detection is the process of identifying and analyzing differences between two or more video frames captured at different times. The goal is to detect meaningful changes in a scene, such as the appearance or disappearance of objects, modifications in the environment, or movement. This technique is crucial in various applications, including surveillance, forensic analysis, and environmental monitoring. For example, in surveillance systems, video change detection can automatically flag when an object is left behind or removed from a scene, such as in cases of suspicious activities. The process typically involves comparing frames pixel by pixel or analyzing patterns in object movements to detect significant alterations. However, challenges such as lighting variations, shadows, and background movement (e.g., trees swaying) can complicate accurate detection. Advanced techniques, like background subtraction, optical flow, and deep learning, help improve the accuracy of detecting only meaningful changes while minimizing false positives caused by noise or minor scene variations.

Consequently, sophisticated deep learning-based techniques are utilized to identify changes between a pair of videos. Collecting sufficient labeled video data for training large deep learning models is time-consuming, cumbersome, and expensive. As such, it is imperative to develop a few-shot, ideally, a zero-shot video change detection technique for industrial applications where labeled data are scarce. Zero-shot change detection refers to a method that identifies changes between two sets of data without requiring any labeled training examples of those changes. In the context of video analysis, the model does not rely on previously labeled data indicating what types of changes to look for. Instead, it detects differences by analyzing the features of objects in the data and identifying new, disappeared, or altered elements directly. This approach allows the model to generalize to unseen scenarios without needing specific prior training for each type of change.

Few studies in the literature introduce deep learning video change detection techniques using publicly available datasets. Nevertheless, these methods require huge labeled video data to train deep learning models from scratch. Furthermore, the existing methods are task specific and, hence, difficult to generalize. Therefore, in this paper, we propose a zero-shot video change detection algorithm utilizing pre-trained deep learning models and conventional image processing and computer vision techniques. More specifically, our proposed technique incorporates the following steps: i) identify matching pairs of frames from the pair of input videos, ii) adjust lighting condition between the matching pairs if necessary, iii) adjust any misalignment between the matching frames, iv) utilize an existing pre-trained deep learning object detection model to identify objects in both the matching frames, and finally v) find nonoverlapped object bounding boxes between each pair of video frames to identify the changes. Our proposed method is easily generalizable by making few changes in the steps mentioned above. We investigate the efficacy of our proposed method on the publicly available Video Database of Abandoned Objects (VDAO) collected in a cluttered industrial environment. Our results suggest the proposed zero-shot video change detection framework shows improved performance compared to an ideal change detection scenario.

2. RELATED WORK AND OUR OBJECTIVE

Several works in the literature propose various methods for solving the change detection problem (Zhang et al., 2022; Li et al., 2022; K. Chen et al., 2022; Daudt et al., 2018). However, the majority of the works focus on the image change detection, and hence, video change detection problem is still largely unexplored albeit numerous practicable industrial applications. Consequently, this work proposes a novel framework for zero shot video change detection in the complex industrial environment.

Image change detection has been an active research area for decades, with numerous techniques proposed over the years. These approaches can broadly be categorized into traditional methods and deep learning-based methods. Traditional approaches involve pixel-based methods, such as image differencing and image ratioing, which calculate the difference or ratio between pixel values from two images to identify changes (Singh, Harrison, & Aggarwal, 1989). Another approach, change vector analysis (CVA), analyzes the difference in spectral bands and principal components of multispectral images to detect changes (Malila, 1980). With the advent of machine learning, researchers began exploring supervised and unsupervised techniques. Supervised methods include support vector machines (SVM), decision trees, and random forests, which require labeled data to learn and classify changes (Bruzzone, Rizzo, Gaddi, & Marconcini, 2004). Unsupervised methods, such as K-means clustering and Gaussian mixture models, identify changes by grouping pixels with similar characteristics (Celik, 2010). Deep learning-based approaches have recently gained popularity in image change detection due to their ability to extract features automatically and model complex relationships. Convolutional neural networks (CNNs) have been widely used for this task, including architectures like U-Net, FC-EF, and ResNet (Zhang et al., 2022; Li et al., 2022; K. Chen et al., 2022). Autoencoders have also been employed for unsupervised change detection, as they can learn meaningful representations from the data without requiring explicit labels (Daudt et al., 2018). Briefly, image change detection has evolved from pixel-based and traditional machine learning methods to deep learning approaches, showcasing the continuous progress in this field. As techniques continue to advance, so does the potential for more accurate and efficient change detection systems.

Video change detection poses greater challenges than image change detection due to various factors, with labeled data scarcity and complexity being key contributors. Labeled data scarcity is a significant issue in video change detection, as large amounts of annotated data are often unavailable or challenging to obtain. This is due to the time-consuming nature of annotating video datasets and the potentially high costs associated with the process. Consequently, unsupervised methods, transfer learning, traditional computer vision approaches and a combination of all of the above approaches may be necessary to design a robust video change detection algorithm. The complexity of video data is another critical challenge in video change detection. Videos contain spatial and temporal dimensions, leading to larger data volumes compared to images. Analyzing temporal information requires capturing spatial changes over time, which can be computationally expensive and challenging to model. Moreover, videos often exhibit different scene variations, camera motions, illumination changes, and occlusions, further complicating the change detection process. Additionally, the presence of irrelevant motions, such as moving background objects or camera jitter, can introduce noise and hinder accurate detection. In summary, video change detection is more challenging than image change detection due to labeled data scarcity and the inherent complexity of video data. Addressing these challenges requires innovative approaches to training models and efficiently handling spatial and temporal information.

Video change detection has evolved significantly over the years, with various approaches proposed to address the challenges posed by the temporal dimension of video data. Traditional video change detection methods typically involve background subtraction techniques, which identify changes by comparing each frame to a background model (Elgammal, Harwood, & Davis, 2000). Techniques such as frame differencing, optical flow, and motion compensation have also been employed for change detection (Hu et al., 2011; Mahadevan & Vasconcelos, 2012). Handcrafted feature-based approaches extract features like histogram of oriented gradients (HOG), scale-invariant feature transform (SIFT), and local binary patterns (LBP) from video frames (Mittal & Zisserman, 2013; Saha, Chaudhury, Banerjee, & Saha, 2013). These features are then analyzed for changes using various machine learning techniques, including support vector machines (SVM) and Gaussian mixture models.Deep learning-based methods have recently emerged as powerful tools for video change detection. Recurrent neural networks (RNNs), such as long short-term memory (LSTM) networks, have been applied to model temporal information in videos (Z. Chen et al., 2018). Convolutional neural networks (CNNs) and their variants, such as two-stream CNNs and fully convolutional networks, have also been used to extract spatiotemporal features (Bai et al., 2019; Yang, He, Chen, Tian, & Yu, 2020). More recently, transformers have demonstrated strong potential in video change detection, as they can efficiently capture long-range dependencies (Zhu et al., 2021). Nevertheless, collecting large amount of labeled video data is very difficult especially in real life industrial use cases. As such, zero-shot video change detection has emerged as a promising approach to address the challenges of annotated data scarcity in video analysis.

Zero-shot video change detection methods aim to detect changes in unseen scenarios without requiring labeled data for specific events or categories. One approach involves leveraging pre-trained visual-language models that have been trained on large-scale image-text datasets (Ahmad et al., 2023). These models learn to associate visual features with textual descriptions, enabling zero-shot recognition of actions or events in videos. Another line of work utilizes foundation models like the Segment Anything Model (SAM) (Telle et al., 2023). In this approach, the model detects semantic regions in previously acquired maps and live views, and change detection is performed by comparing the segmentation masks. Additionally, some techniques such as event composition knowledge extracted from web images (Gan et al., 2017) are utilized for zero-shot event detection in videos. These methods aim to understand the relationships between events and recognize unseen events based on their composition. However, the above mentioned methods suffer computational complexity and generalization issues. Furthermore, the foundational visual-language models are resource intensive which is prohibitive in most real life industrial applications.

In view of the challenges posed by generalization and resource-intensive nature of existing methods, this research presents a robust zero-shot video change detection framework specifically designed for real-life industrial applications. Our novel approach overcomes these obstacles by integrating pre-trained deep learning models with conventional image processing and computer vision techniques. The key steps of our method include:

1. Matching frame identification from a pair of input videos.
2. Lighting condition adjustment between the matching frames.
3. Misalignment correction between the matching frames.
4. Object detection in both frames using a pre-trained deep learning model.
5. Non-overlapping object bounding box comparison to identify changes.

Our proposed technique can be easily generalized with minor adjustments in the above steps.

3. BACKGROUND

The following few sections provide a brief background on the YOLO object detection model, color transfer and video frame matching techniques.

3.1. YOLO Object Detection

YOLO is a real-time object detection model that balances speed and accuracy. It combines the efficiency of a unified architecture with the precision of a specialized detection method, making it effective across various applications and deployment scenarios. In this work, we utilize the more advanced YOLOv7 (Cholakkal et al., 2022) object detection model. The YOLOv7 model consists of scaled-YOLOv7 (S-YOLOv7) and CSP-YOLOv7, which enable the model to significantly outperform other state-of-the-art detectors. S-YOLOv7 introduces architectural changes such as scaling the YOLOv7 architecture to different sizes, allowing for efficient and accurate object detection across various devices and datasets. This adaptability ensures the model's performance is maintained even when deployed on resource-constrained devices. CSP-YOLOv7 is another important component of the overall architecture, leveraging the Cross-Stage-Partial (CSP) approach to further improve the model's performance. By implementing CSP modules within the YOLOv7 framework, CSP-YOLOv7 enhances the extraction and processing of essential object features, leading to increased detection accuracy. The YOLOv7 architecture is trained from scratch on the MS COCO dataset, demonstrating its effectiveness in handling diverse and complex object detection tasks. We utilize the pre-trained YOLOv7 model to solve the object detection step of our proposed video change detection algorithm. However, in scenarios where the object of interest in the video is unique or significantly distinct from the objects used to train YOLOv7, finetuning the model with a small custom-labeled dataset may be necessary for optimal performance.

3.2. Color Transfer

Image color transfer aims to modify a target image's colors to match the color palette, tone, or lighting of a reference image. Common methods include palette-based clustering for segmenting images into color regions and defining color mapping strategies to reproduce the desired color distribution. Additional techniques like histogram matching, neural style transfer, and lighting optimization enhance the visual similarity between reference and target images for seamless color transfer.This paper employs a fast and robust color transfer method proposed by (Reinhard, Adhikhmin, Gooch,

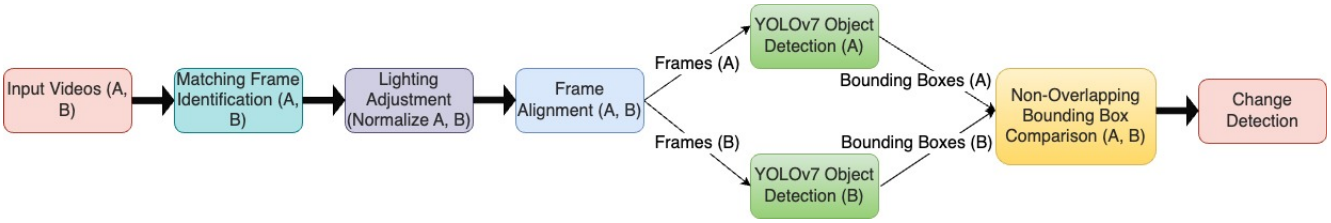
Input Videos (A, B)
Matching Frame Identification (A, B)
Lighting Adjustment (Normalize A, B)
Frame Alignment (A, B)
Frames (A)
Frames (B)
YOLOv7 Object Detection (A)
YOLOv7 Object Detection (B)
Bounding Boxes (A)
Bounding Boxes (B)
Non-Overlapping Bounding Box Comparison (A, B)
Change Detection

Figure 1. Overall pipeline of the zero-shot video change detection framework.

& Shirley, 2001), which effectively transfers color characteristics between images, such as color palette, tone, and lighting. The approach utilizes palette-based clustering to segment images into different color regions and defines a color mapping strategy to replicate the reference image's color distribution onto the target image. Lighting optimization is also applied to further improve the visual similarity between the two images, resulting in a more seamless color transfer.

3.3. Video Frame Matching

Video frame matching (Teller, Larsen, & Foga, 2022; Laptev, Weickert, & Rae, 2022) is a process that involves establishing spatial and temporal correspondences between frames in a video sequence or between frames from different videos. The goal is to correct any misalignments caused by camera motion, object movement, or variations in the scene, ensuring that frames can be accurately compared or combined for tasks such as video stabilization, object tracking, or video similarity analysis. Common methods include feature-based matching, optical flow estimation, and deep learning-based techniques for accurate frame registration and alignment. In this paper, we utilize a ResNet based feature matching algorithm to solve the video frame matching step in our proposed video change detection framework.

4. METHODOLOGY

Figure 1 presents an overview of the proposed zero-shot video change detection framework, which comprises several critical components designed to address real-world challenges, such as frame misalignments and varying lighting conditions. The framework consists of the following steps.

i) Frame Matching - Establishing Corresponding Pairs

The inputs to the video change detection framework are a pair of video sequences. Video sequences often suffer from frame misalignments, making it difficult to directly compare corresponding frames from video A and video B. To address this challenge, this framework employs a ResNet-based feature matching algorithm to identify the most similar frames between videos A and B. Specifically, the algorithm extracts frames from both videos at regular intervals, preprocesses them by resizing and normalizing, and then computes feature vectors using the ResNet18 model. These vectors capture high-dimensional representations of the frames' visual content. The cosine similarity between feature vectors of frames from the two videos is calculated to find the most similar pairs. This method selects the top n pairs based on their similarity scores. The chosen features and distance metric used to compare them determine the sensitivity of the matching process to factors such as lighting variations and camera motion. By accurately matching frames despite these challenges, the framework establishes a robust foundation for subsequent change detection. This ensures that any differences detected between the videos are based on relevant, corresponding frames, enhancing the reliability of the comparison. Therefore, the initial step guarantees that our method remains unaffected by the varying number of frames between the two videos, discrepancies in their starting points, and differences in the frame rates of the videos.

ii) Lighting Adjustment - Ensuring Consistency

Videos A and B might be captured under different lighting conditions, leading to inconsistencies that can hinder change detection. The framework utilizes an efficient color transfer technique proposed by (Reinhard et al., 2001) to mitigate this challenge. This process adjusts the lighting of the matched frame from video B to align it with the lighting of the corresponding frame from video A. By mitigating lighting inconsistencies, the framework allows for more accurate object detection and change identification in the following steps.

iii) Frame Alignment - Refining Correspondence

Despite successful matching, minor misalignments may still occur within frames due to factors such as camera motion or object movements. This necessitates the calculation of horizontal and vertical shifts between the matched objects within the frames. To address this, various techniques can be employed to align the images. One approach involves leveraging geometric methods to utilize the scene's geometry for alignment. For instance, homography estimation techniques can be utilized to determine the transformation matrix that relates corresponding points in the two images, assuming that the scene is planar. Additionally, other methods exploit the epipolar geometry between the images to establish correspondence and align them. Furthermore, techniques like normalized cross-correlation between local image patches can be employed to determine the optimal horizontal and vertical

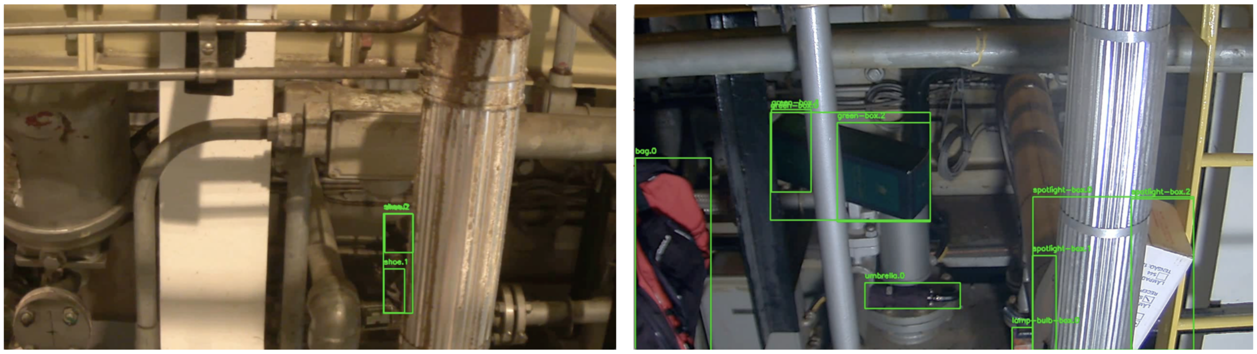

Figure 2. Example of labeled VDAO dataset, left is single object, right is multiple objects.

shifts for alignment. By applying the calculated shifts, the framework aligns the B frames with the A frames, ensuring precise object localization across matched frames and leading to more reliable change detection. This comprehensive approach to frame alignment is essential for ensuring the accuracy and reliability of image processing tasks.

iv) Object Detection - Identifying Objects of Interest

This step leverages a pre-trained or fine-tuned YOLOv7 object detection algorithm to identify objects within both the original A frame and the aligned B frame. YOLOv7 is a state-of-the-art object detection algorithm known for its ability to detect objects in real time with high accuracy. It predicts bounding boxes and corresponding object class labels for each detected object within an image. Object detection provides crucial information about the presence and location of objects in each frame, allowing the framework to pinpoint potential changes.

v) Identifying Changes - Analyzing Overlaps

The final step focuses on analyzing the overlaps between the bounding boxes identified in the A and B frames using object detection technique mentioned in the previous step. The framework calculates the intersection area between each pair of bounding boxes. A high degree of overlap suggests the object might be present in both frames, while minimal or no overlap indicates a potential change. Bounding boxes with minimal or no overlap in the A and B frames are flagged as potential new objects or areas of change. This information can be further processed to pinpoint specific types of changes, such as object additions, removals, or modifications. Techniques like analyzing object class labels or comparing image patches within non-overlapping bounding boxes can be employed to refine change detection and potentially identify the nature of the change (e.g., object type change, damage assessment).

In summary, our proposed zero-shot video change detection framework offers a robust and efficient solution to identify changes in video sequences. By combining frame matching, lighting adjustment, object detection, and change identification algorithms, the proposed pipeline effectively addresses real-world challenges, such as frame misalignments and varying lighting conditions, ensuring reliable performance in practical scenarios. It is important to highlight that our model is referred to as zero-shot because it does not rely on labeled changes within the videos. Specifically, in this paper, we do not use labeled abandoned objects to perform the change detection task; labels are only employed for performance evaluation. Instead, our approach identifies objects in both videos and compares the overlap in their detections to detect new objects.

5. RESULTS AND DISCUSSIONS

5.1. VDAO Dataset Description

The authors in (Freitas et al., 2014) presents a comprehensive video database called VDAO, which is designed to evaluate surveillance systems for the automatic detection of abandoned objects in cluttered environments. The dataset provides two types of videos, i) a reference video of the cluttered environment without any abandoned object which we refer as $Video_{Ref}$ and ii) a video of the same cluttered environment with the addition of the abandoned object(s) which we refer as $Video_{Obj}$. More comprehensively, the dataset includes 66 videos: 56 single-object shown in the left of Figure 2, 6 multi-object shown in the right of Figure 2, and 4 reference videos without objects, recorded under two lighting conditions (spotlight and natural light) using two high-resolution cameras. The single-object videos feature 9 different objects, each captured in 3 distinct positions, while the multi-object videos consist of 15 objects placed in 3 different arrangements. This results in an approximate total of 8.2 hours of video. Each frame in the dataset is meticulously annotated with bounding boxes around the objects. In our work we utilize this dataset to evaluate the performance of our proposed zero-shot video change detection framework.

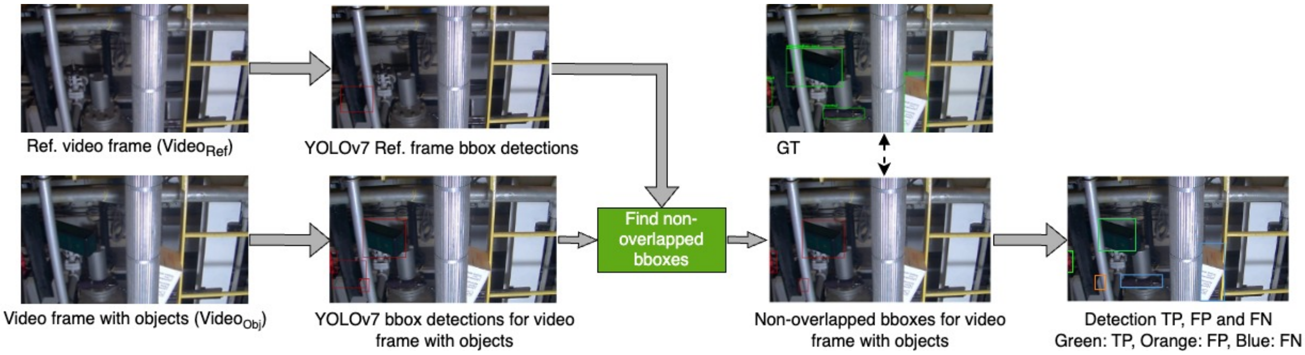
Ref. video frame (Video$_{Ref}$)
YOLOv7 Ref. frame bbox detections
GT
Video frame with objects (Video$_{Obj}$)
YOLOv7 bbox detections for video frame with objects
Find non-overlapped bboxes
Non-overlapped bboxes for video frame with objects
Detection TP, FP and FN
Green: TP, Orange: FP, Blue: FN

Figure 3. Example of proposed video change detection pipeline. Notably, in the final output majority of the FPs are removed by the non-overlap bounding box identification algorithm.

5.2. Experiment Setup and Evaluation

The objects used in the VDAO dataset are obscured in an industrial setting. Therefore, directly applying a pre-trained deep model is challenging. As such, we fine-tune a pre-trained YOLOv7 model using the objects in the VDAO dataset videos. As mentioned in the previous section, the VDAO dataset includes 56 object classes, such as *white-box*, *dark-blue-box*, *black-backpack*, and *pink-backpack*. We reduce the class granularity by grouping the object classes into 13 classes: *backpack*, *box*, *bottle*, *coat*, *bag*, *bottle-cap*, *mug*, *string-roll*, *umbrella*, *wrench*, *jar*, *shoes*, and *towel*. We randomly sample 500 frame images from all classes to fine-tune YOLOv7 model for 50 epochs. We ensure that the training videos utilized for object detection fine-tuning task are separate from the test videos.

The main objective of this work is to detect the changes between the reference ($Video_{Ref}$) and the videos with objects ($Video_{Obj}$). The video change detection task is obtained by following the pipeline show in Figure 1. For better understanding, an example of the major steps of our proposed algorithm using the videos from the VDAO dataset is shown in Figure 3. We evaluate the performance of our proposed method by computing the frame level object detection accuracy. For each pair of frames we run the fine-tuned YOLOv7 object detection model on both the matched frames of $Video_{Ref}$ and $Video_{Obj}$. The output of the YOLOv7 model is a set of bounding boxes obtained from $Video_{Ref}$ frame and the $Video_{Obj}$ frame. We apply non-overlapping bounding box identification algorithm on the set of $Video_{Ref}$ frame and the $Video_{Obj}$ frame bounding boxes. We compare the final outcome with the ground truth to identify which bounding boxes are true positives (TP), false positives (FP) and false negatives (FN). We compute TP, FP, and FN for all the frames of the $Video_{Ref}$ and $Video_{Obj}$. We repeat this process for six different videos of three different object categories to obtain the total number of TP, FP, and FN. Finally, we compute the precision and recall metrics using the total number of TP, FP, and FN.

The existing deep learning based video change detection techniques are trained using labeled data, and hence, are unsuitable to perform a direct comparison with our proposed method. As such, we utilize an ideal scenario where we assume that it is already known which frames of the $Video_{Obj}$ contains the new objects. In this case the $Video_{Ref}$ video is ignored to identify the change. We run the fine-tuned YOLOv7 model only on those frames to obtain the object detection outputs. We then compare the detection outputs with the ground truth to obtain TP, FP, and FN. The pipeline of the ideal change detection case in shown in Figure 4. In this case also we utilize the exact same six videos that we used for our proposed method. Similarly, for the ideal pipeline we compute the precision and recall metrics using the total number of TP, FP, and FN.

For the videos containing multiple objects, we follow the same evaluation strategy. However, the VDAO dataset includes limited videos containing multiple objects. We use two videos which provides ground truth labels for performance evaluation.

5.3. Analysis of Results

We first evaluate the results for the videos containing single objects. Table 1 shows a comparison between the ideal change detection case and our proposed zero-shot change detection technique for the single object video use case in terms of TP, FP, FN, Precision and Recall. We use two different intersection of union (IOU) thresholds to determine correctness of a bounding box. For example, an IOU threshold greater than 0.5 means, if there is a 50% overlap between the predicted bounding box and the ground truth box we consider that bounding box as a correct detection i.e. TP. Understandably, setting a higher IOU threshold results in poor correct bounding box detection performance. Table 1 demonstrates that our proposed method shows slightly better performance compared to the ideal scenario when IOU threshold is greater than 0.5. However, when the IOU threshold is greater than 0.95 our method performs significantly better than the ideal

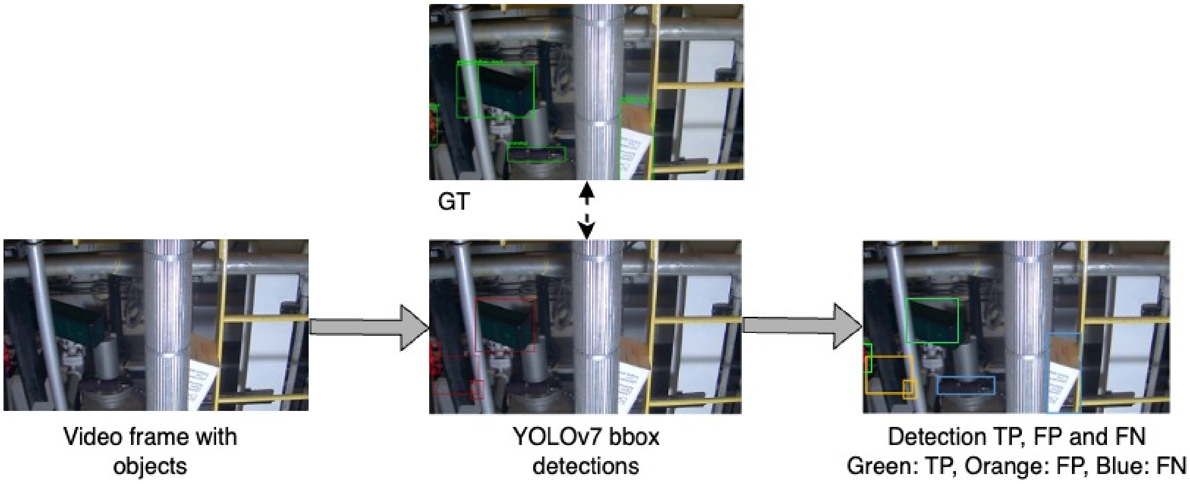

Figure 4. Ideal change detection pipeline. It is assumed that the pipeline already has access to the frames where the new object appears without using the reference video frames.

scenario. It should be noted here that our proposed algorithm automatically reduces many false positives during the non-overlap bounding box identification steps, and hence, the overall performance improvement.

	Ideal Change Detection	**Our Proposed**
$IOU > 0.5$		
TP	4158	4156
FP	198	90
FN	12	12
Precision	0.95	0.98
Recall	0.996	0.997
$IOU > 0.95$		
TP	966	966
FP	3390	2760
FN	3204	3198
Precision	0.22	0.26
Recall	0.231	0.232

Table 1. Comparison between ideal change detection scenario vs our proposed method for single object video use case.

Next we use the videos containing multiple objects to compare our proposed method with the ideal change detection case. Table 2 shows the comparison between the two methods in terms of TP, FP, FN, Precision and Recall for two different IOU thresholds, 0.5 and 0.95, respectively. In this case also, the results suggest that our proposed algorithm shows significantly improved performance compared to the ideal case scenario. Once again, Table 2 demonstrates the FP reduction efficacy of our proposed algorithm.

It is observed from Table 2 that the object detection performance for the multi object use case is significantly lower than that of the single object use case. The number of FPs and FNs are significantly higher in this case. This is due to the hidden positioning and small size of the objects. Figure 5 shows an example where some target objects are hidden in the background, making them difficult to detect even with the human eye. These objects are highlighted with orange bounding boxes. Furthermore, we fine-tuned the YOLOv7 model using only 500 images in total across all classes. Additionally, the number of samples for some classes are very few in the training set.

	Ideal Change Detection	**Our Proposed**
$IOU > 0.5$		
TP	9995	9985
FP	4275	1821
FN	5802	5817
Precision	0.70	0.85
Recall	0.63	0.63
$IOU > 0.95$		
TP	1786	1794
FP	8277	2760
FN	9520	9516
Precision	0.17	0.39
Recall	0.16	0.16

Table 2. Comparison between ideal change detection scenario vs our proposed method for multiple object video use case.

6. CONCLUSION

This paper presents a zero-shot video change detection algorithm which leverages pre-trained deep learning models, conventional image processing and computer vision techniques to detect changes between pairs of input videos. Our proposed method is designed to be easily generalizable by making few changes, allowing it to be applied in various industrial applications where labeled data are scarce. Through extensive experimentation publicly available VDAO change detection dataset collected in a cluttered industrial environment,

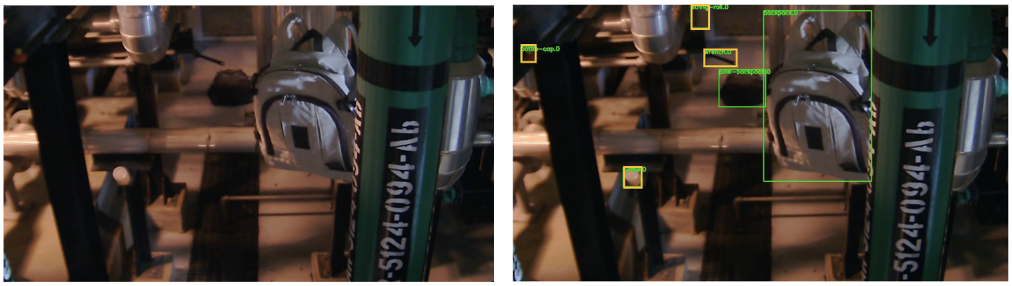

Figure 5. Example of some of the objects hidden in the background. Left is the object video, Right is the ground truth for that frame, hidden object is highlighted in orange.

we demonstrate the efficacy of our proposed change detection method. The VDAO dataset comprises of videos containing single and multiple objects. The change detection task involves identifying the changes in the videos compared to a reference video that may or may not contain any object. Our results suggest that our proposed zero-shot video change detection method shows improved performance compared to an ideal change detection scenario. While our approach leverages the strengths of pre-trained object detection models, its success relies heavily on the accuracy of the object detection models; therefore, any limitations in the object detection model performance may have a ripple effect and impact the overall performance of the change detection technique. In the future, we plan to further evaluate our proposed method on more datasets and explore its application in various real-world scenarios.

References

Ahmad, F., Shan, Z., Wang, K., You, S., Zhou, B., Gu, S., … Xu, Z. (2023). Ez-vcd: Efficient zero-shot video change detection with pretrained visual-language models. *arXiv preprint arXiv:2303.01339*.

Bai, X., Ma, C., Li, Y., Guo, J.-J., Xia, Z., & Guo, H. (2019). Video change detection with fully convolutional networks. *Multimedia Tools and Applications*, *78*(16), 23205–23230.

Bruzzone, L., Rizzo, D., Gaddi, L., & Marconcini, M. (2004). A novel approach to the selection of spatially invariant features for change detection in high-resolution images. *Pattern Recognition Society, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer*, *1*, I–I.

Celik, T. (2010). Unsupervised change detection in satellite images using principal component analysis and k-means clustering. *IEEE Geoscience and Remote Sensing Letters*, *7*(4), 739–743.

Chen, K., Zhang, B., Yang, Z., Che'n, Y., Zhong, Y., Li, G., … Li, H. (2022). A survey of deep learning-based remote sensing image change detection. In *Remote sensing* (Vol. 14, p. 4630).

Chen, Z., Xu, Y., Wang, C., Chen, Q., Yang, X., et al. (2018). Real-time video change detection using deep-learning based models. *arXiv preprint arXiv:1810.00563*.

Cholakkal, A. A., Jinek, S., Qi, Y., Jiang, W., Cholakkal, A., Ramakrishnan, S., & Davis, L. S. (2022). *Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.*

Daudt, R., Kleine, R., Glasmachers, T., Ball, L., Rosendahl, P., & Wesemann, L. (2018). Deep transfer learning for efficient image change detection. In *arxiv preprint arxiv:1805.12552.*

Elgammal, A., Harwood, D., & Davis, L. (2000). Non-parametric model for background subtraction. *European Conference on Computer Vision*, 751–767.

Freitas, G., Lopes, N., Jorge, R., Viana, A., Pontes, B., & Ribeiro, R. (2014). An annotated video database for abandoned-object detection in a cluttered environment. In *International telecommunications symposium (its), 2014* (pp. 1162–1166).

Gan, C., Feng, Y., Blasch, E., Shen, J., Zhang, W., Le, J.-B., … others (2017). Deck: Deep event composition knowledge for zero-shot event detection in video. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2072–2081.

Hu, Y., Guo, Y., Xu, Y., Liu, Z., Yuan, Y.-P., et al. (2011). Video change detection based on frame differencing and gaussian mixture modeling. *Pattern Recognition Letters*, *32*(14), 1727–1734.

Laptev, I., Weickert, M. H. S., & Rae, M. (2022). Multiscale video sequence matching for near-duplicate detection and localization. *IEEE Transactions on Multimedia*, *24*(2), 1151–1162.

Li, F., Zhang, B., Han, X., Chen, K., Zhang, G., & Wu, Y. (2022). Remote sensing image change detection with transformers. In *arxiv preprint arxiv:2210.09272.*

Mahadevan, V., & Vasconcelos, N. (2012). Scene change detection: A survey. *Pattern Recognition*, *46*(1), 398–408.

Malila, W. A. (1980). Change detection in urban areas using multispectral aerial photography. *Proceedings of the ESA/JRC Workshop on Remote Sensing for Environmental Monitoring of Urban Areas*, 1–8.

Mittal, A., & Zisserman, A. (2013). Video change detection using optical flow based descriptors. *International Conference on Computer Vision*, 455–462.

Reinhard, E., Adhikhmin, M., Gooch, B., & Shirley, P. (2001). Color transfer between images. *IEEE Computer Graphics and Applications*, *21*(5), 34–41.

Saha, S., Chaudhury, K., Banerjee, A., & Saha, B. (2013). Change detection in video using local binary pattern. *International Journal of Engineering Research & Technology*, *2*(5), 1462–1467.

Singh, A., Harrison, J., & Aggarwal, K. (1989). Image differencing and monitoring urban land-use change. *Photogrammetric Engineering and Remote Sensing*, *55*(10), 1357–1368.

Telle, L., Liao, H.-T., Tao, H., Xu, A., Zha, H., Tokmakov, P., … others (2023). Lasersam: Scene-aware semantic annotation of 3d point clouds. *arXiv preprint arXiv:2303.07930*.

Teller, S. B., Larsen, K., & Foga, M. (2022). Video matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(3), 1443–1456.

Yang, M., He, Q., Chen, Y., Tian, Q., & Yu, D. (2020). Video change detection with a two-stream convolutional neural network. *Neurocomputing*, *403*, 341–348.

Zhang, B., Guo, H., Chen, K., Li, S., Sun, K., Li, J., … Wu, Y. (2022). Maskcd: A remote sensing change detection benchmark with annotation masks. In *arxiv preprint arxiv:2212.05316*.

Zhu, L., Zhan, H., Liu, C., You, F., Qin, R., et al. (2021). Video change detection with transformers. *arXiv preprint arXiv:2102.12720*.