Integrating Spatial Proximity and Visual Feature Similarity for Crew Group Detection in Construction Site

Cheng-Yun Tsai[1], Jacob J. Lin[1], Ci-Jyun Liang[2]

[1]Department of Civil Engineering, National Taiwan University, Taiwan
[2]Department of Civil Engineering, Stony Brook University, USA

r12521709@ntu.edu.tw, jacoblin@ntu.edu.tw, Ci-Jyun.Liang@stonybrook.edu

Abstract -

Crew-level productivity analysis plays a crucial role in construction site management, as it provides a macro-level understanding of workforce performance. While traditional productivity measurement methods, such as work sampling, group timing technique, and five-minute rating, offer valuable insights, they rely heavily on manual observation, making them labor-intensive and prone to inaccuracy. In this study, we propose a deep learning-based framework for automated crew-level identification. The framework employs a graph-based approach that integrates visual feature similarity and spatial proximity of workers, combined with clustering algorithms, to detect and analyze worker groups. The proposed method is validated on a construction site dataset collected from a rebar installation task. Experimental results demonstrate the framework's effectiveness, achieving high accuracy in group detection with robust performance across various evaluation metrics. This work highlights the potential of automated systems to enhance construction site management by reducing reliance on manual observation and providing real-time insights into crew-level productivity.

Keywords -

Group Detection; Clustering; Image Understanding; Human Activity Recognition

1 Introduction

Labor resources play a critical role in the success of construction projects [1]. However, construction sites typically consist of distinct groups of workers collaborating to accomplish tasks. From a macro perspective, analyzing productivity at the crew level—rather than focusing on the productivity of individual workers—provides a more comprehensive view of progress tracking and performance analysis [2]. By incorporating additional contextual information such as objects or spatial locations, this analysis can help managers identify other potential causes of delays, such as inefficient material placement or work environment constraints. Therefore, understanding and monitoring crew-level productivity is essential for improving construction site management.

Traditionally, three primary methods have been used for productivity measurement on construction sites: work sampling, group timing technique, and five-minute rating [3]. Work sampling involves observing workers at random intervals to determine the percentage of time spent on productive and non-productive activities, providing insights into overall labor efficiency. The group timing technique, on the other hand, focuses on measuring crew cycle times by observing work processes at fixed intervals, which allows managers to assess the efficiency of group operations and evaluate the appropriateness of crew size. Finally, the five-minute rating is a simpler and faster approach that estimates delay times by periodically observing crew activities over short durations, such as five minutes. While these methods are effective for gathering productivity data, they rely heavily on manual observation, which is labor-intensive, time-consuming, and dependent on the experience and judgment of observers.

Recent advancements in technology, particularly the widespread deployment of cameras on construction sites and developments in deep learning and computer vision, have made automated productivity monitoring increasingly feasible. These automated systems address the limitations of traditional methods by reducing the need for labor-intensive and time-consuming manual observation. Unlike methods that rely heavily on the observer's experience and continuous monitoring, automated systems offer real-time and accurate productivity tracking. This technological shift enables site managers to focus their efforts on more strategic aspects of project management while ensuring the reliability and efficiency of data collection.

In recent studies, significant progress has been made in motion recognition for individual workers. Luo et al. [4] propose an improved CNN integrating RGB, optical flow, and gray stream data to accurately recognize and assess a single worker's reinforcement installation activity in videos. Luo et al. [5] used spatial and temporal convolutional networks with a novel fusion strategy to recognize multi-workers' actions from surveillance videos. However, these studies primarily focus on the productivity of individual workers and lack an in-depth discussion of

crew-level performance.

A limited number of studies have explored the relationships between workers and workers or workers and objects. The same author from [5] proposes a two-step method, which uses CNN to detect objects and workers and then analyzes semantic and spatial relevance between each entity [6]. Cai et al. [7] introduce a two-step classification method combining working group identification and activity recognition using LSTM networks, leveraging pre-computed positional cues (eq. bounding box overlapping) and attentional cues (eq. head pose, body orientation) to recognize construction activities with multiple entities. Although these studies successfully identify the relationships between objects in construction site images, their methods for worker group identification remain constrained by precomputed handcrafted features or site-specific rules established by domain experts. Therefore, developing an effective method for identifying and analyzing worker groups on construction sites remains a significant research challenge.

Building on these insights, this study contributes to the field by proposing a deep learning-based framework for automated crew-level identification. Unlike previous methods that rely on handcrafted features or predefined spatial heuristics, our approach dynamically integrates visual feature similarity and spatial proximity into a unified relation matrix. This enables the model to adaptively cluster workers based on real-world interactions rather than being constrained by rigid rules or single-modality input. Additionally, this method simplifies the system's deployment requirements, as it only requires image input and a person detection model, eliminating the need for manually predefined spatial constraints or external tracking devices. The framework is validated using a dataset collected from a rebar installation task on a construction site, demonstrating its robustness across diverse scenarios.

2 Methodology

In this study, our objective is to cluster workers in the scene into different groups; we combine both spatial relevance and feature relevance to represent the affinity matrix between the workers in the scene. Figure 1 shows the overview of our framework, which contains four major modules: (a) Feature Extraction, (b) Graph Construction, (c) Spatial Distance Encoding with Masking, and (d) Relational Matrix for Clustering. We will elaborate on the details of these four modules in the following section.

2.1 Feature Extraction

The purpose of this module is to extract the great information of visual feature reliance in the image for each worker. As the main goal of this research is to cluster the workers, we assume that there will be other modules to complete the object detection of workers. Therefore, the inputs of the proposed framework are the RGB image with each worker's bounding box. We first utilize Inception-v3 [8], a CNN-based backbone model combined with ROI-Align [9], to extract spatially precise feature representations for each entity.

In this study, we use Inception-v3 [8] as the feature extraction backbone. To enhance robustness and leverage prior knowledge, we initialize the model with pretrained weights from the Collective Activity Dataset [10], which has been optimized for human-centric activity recognition. This choice allows us to extract meaningful features while mitigating the risk of overfitting. Given the focus of our study on clustering methods rather than CNN architecture comparisons, we selected a well-established backbone known for its balance between efficiency and accuracy in feature extraction.

ROI-Align [9] is a region-based operation designed to extract features corresponding to a specific region of interest (ROI) from the feature maps produced by the CNN backbone. Unlike earlier methods, such as ROI-Pooling, ROI-Align avoids quantization errors by using bilinear interpolation to precisely map the input pixels to the feature space.

2.2 Graph Construction

After extracting the feature from the image, we want to calculate the affinity further according to the similarity of visual features. We can represent the individuals in any scene as a graph, where each node represents the individuals and the edge represents the connection or the relationship between these two nodes. By leveraging this graph structure, we aim to compute the visual feature affinity matrix R_E, which quantifies the affinity between features.

The graph can be denoted as $G = (V, E)$, where V represent the vertices and E represent the edges, and since we are not sure of the exact relationship in the scene in present, there might be any possible pairs of relationships between each worker, so we construct a fully connected graph; the number of vertices N is equal to the number of workers in the scene, and the number of edges is equal to $\binom{N}{2}$. To represent the relationship $e_{u,v}$ between two entities u and v, we adopt the approach introduced in [11]:

$$e_{u,v} = F_1(f_u) \cdot (F_2(f_v))^T, \quad u, v \in \mathcal{V}, \tag{1}$$

where f_u and f_v are the feature vectors of entities u and v, respectively. F_1 and F_2 are learnable linear transformations that project the feature vectors into a relational feature space. The resulting value $e_{u,v}$ represents the connection strength or similarity between u and v. The complete **feature affinity matrix $\mathbf{R_E} \in \mathbb{R}^{N \times N}$**, where N is the number

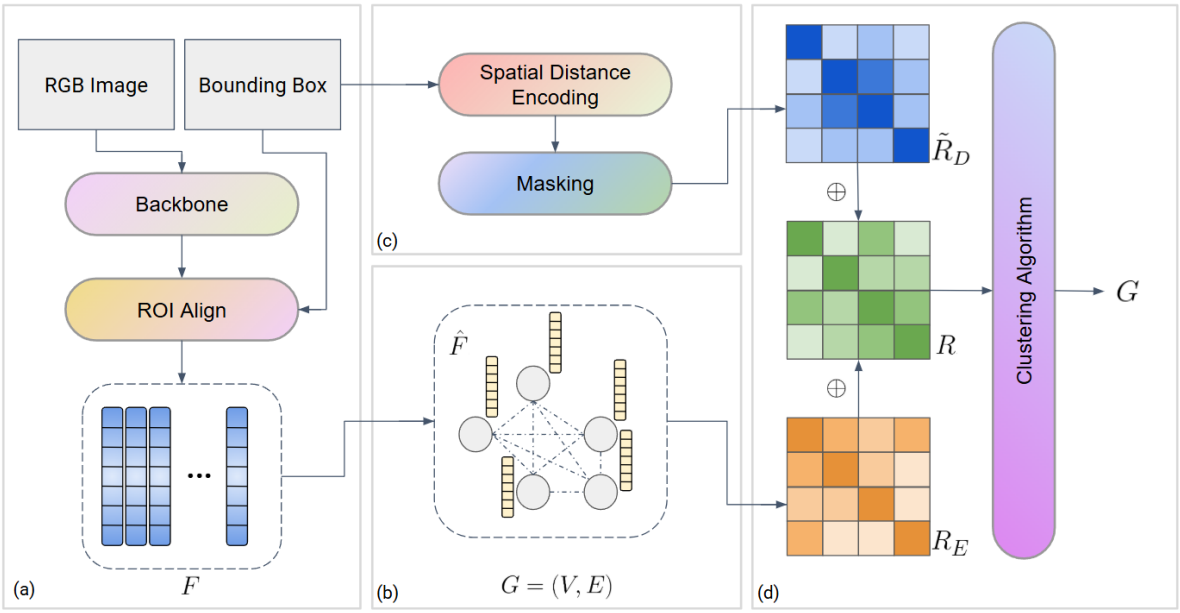


Figure 1. Overview of the Proposed Framework: Key Components – (a) Feature Extraction: extracts high-level features from RGB images and corresponding bounding boxes. (b) Graph Construction: builds a graph structure to represent the extracted features and their relationships, where nodes correspond to features and edges indicate potential connections. (c) Spatial Distance Encoding with Masking: encodes spatial distances and applies masking strategies to enhance the relevance of nearby entities while filtering out irrelevant ones. (d) Relational Matrix for Clustering: computes a relational matrix to facilitate feature grouping and clustering based on the encoded relationships.

of nodes, is constructed as:

$$\mathbf{R_E} = \left[e_{u,v}\right]_{u,v\in\mathcal{V}}. \tag{2}$$

To ensure the weights are normalized, we apply a **softmax operation** to each row of the affinity matrix $\mathbf{R_E}$. This normalization ensures that the sum of the affinity weights for each node (row) equals 1. Mathematically, this process is expressed as:

$$e_{u,v} \leftarrow \frac{\exp(e_{u,v})}{\sum_{v'\in\mathcal{V}} \exp(e_{u,v'})}. \tag{3}$$

This softmax operation converts raw edge weights into probabilities.

2.3 Spatial Distance Encoding & Masking

After we have the feature affinity matrix R_E, We also want to include the relationship between workers in space as a reference. The spatial affinity matrix R_D is computed to encode the physical proximity between entities using their Euclidean distance $D(u, v)$. It is defined as:

$$R_D(u, v) = \exp\left(-\frac{D(u, v)}{\alpha}\right), \tag{4}$$

where:

$$D(u, v) = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2}. \tag{5}$$

In these equations, (x_u, y_u) and (x_v, y_v) denote the spatial coordinates of entities u and v, typically represented as the midpoint of the bottom edge of their bounding boxes, $\alpha > 0$ is a scaling parameter that determines the sensitivity of the affinity score to the distance.

The inverse exponential function R_D ensures a smooth mapping of distances to affinity scores, where entities that are spatially close (smaller $D(u, v)$) receive higher affinity values, indicating stronger proximity-based interactions, while entities that are farther apart (larger $D(u, v)$) are assigned smaller affinity values, reflecting weaker spatial relationships.

We further apply a **masking technique** to filter out weak spatial relationships. Specifically, for each element in the spatial affinity matrix R_D, if the Euclidean distance value is smaller than a predefined threshold τ, the corresponding entry is set to zero. Mathematically, the masked spatial affinity matrix $\tilde{R}_D$ is given by:

$$\tilde{R}_D(u, v) = \begin{cases} R_D(u, v), & \text{if } D(u, v) \geq \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Here, τ is a threshold value that controls the sparsity of the affinity matrix.

2.4 Relation Matrix

To quantify relationships between entities, we define a relation matrix R, which integrates both feature-based and spatial relationships to provide a comprehensive representation of interactions. This matrix is composed of the feature affinity matrix R_E and the masked spatial affinity matrix $\tilde{R}_D$, which represents the spatial relationships between entities based on their positions.

Finally, the combined relation matrix R integrates R_E and $\tilde{R}_D$ to capture both visual feature similarity and spatial proximity. It is calculated as:

$$R = \lambda R_E \oplus (1 - \lambda)\tilde{R}_D, \tag{7}$$

where $\lambda \in [0, 1]$ is a weighting coefficient controlling the relative importance of R_E and R_D, and $\oplus$ denotes element-wise addition of the two matrices. With the relation matrix, we can get the group division results through a post-processing method, e.g., a clustering algorithm.

To achieve robust group clustering, we employ the Spectral Clustering algorithm [12], which effectively operates on the constructed relation matrix R. Unlike traditional K-Means, which assumes spherical clusters and requires predefining the number of clusters, Spectral Clustering leverages the graph Laplacian eigenvectors to segment data based on intrinsic connectivity, making it more suitable for identifying worker groups in complex construction site scenarios. Additionally, given our method's graph-based nature, Spectral Clustering provides a more effective approach to clustering workers based on both feature and spatial affinities.

2.5 Loss Function

The loss function is designed to supervises the accuracy of group detection by evaluating the predicted relationship matrix R_{pred} against the ground-truth relationship matrix R_{gt}. Both matrices are of size $N \times N$, where N is the total number of entities. Each element $R_{i,j}$ in the matrix represents the likelihood of a relationship between entities i and j.

The loss is formulated using cross-entropy, applied element-wise across the matrices:

$$\mathcal{L}_{\text{det}} = -\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(R_{\text{gt}}^{(i,j)}\log R_{\text{pred}}^{(i,j)} + \left(1 - R_{\text{gt}}^{(i,j)}\right)\log\left(1 - R_{\text{pred}}^{(i,j)}\right)\right). \tag{8}$$

Table 1. Average and Total Statistics for Frames and Entities

	Avg	Total
Frame	50	608
Entity	5.28	3211

where $R_{\text{gt}}^{(i,j)}$ is the ground-truth value for the relationship between entities i and j (binary: 1 if related, 0 otherwise).$R_{\text{pred}}^{(i,j)}$ is the predicted probability of the relationship between entities i and j.

3 Experiment

In this section, we elaborate on the details of our proposed dataset, the metrics used to evaluate the performance of our framework, and the results of our experiments. Additionally, we provide an in-depth discussion of the findings.

3.1 Dataset

The dataset consists of 13 videos, capturing real-world construction site activities. The videos were recorded under both sunny and cloudy conditions, ensuring coverage of different natural lighting environments. While the camera angles were not entirely consistent across all videos, the majority were captured from a top-down perspective, typically from elevated positions overlooking the construction site. To evaluate the proposed framework in general construction site, we chose to record the video during the rebar installation work item since there are a lot of workers passing between the scene and different groups of workers working simultaneously. Table 1 shows the detailed statistics of the dataset. On average, each video contains 5.2 workers per frame, reflecting realistic worker density variations. The videos were then set to a resolution of 1920×1080, and downsampled to 1fps.

For group annotation, A group of workers who interact with each other or are working on the same object will be considered a group. Figure 2 shows an annotation sample, bounding boxes of the same color in the picture with red lines between them represent that they belong to the same group.

3.2 Metrics

In this section, we discuss how to evaluate the performence of our framework; the task that our framework does can be classified as **Group Detection Task**. Following [11], we use the **Helf Matrix** [13] to evaluate the accuracy of detected groups. The Helf Matrix quantifies the overlap between a ground truth group and a discovered group using the following formula:

Figure 2. Annotation sample

$$\text{Helf Ratio} = \frac{\text{Number of individuals in the intersection}}{\text{Number of individuals in the union}},$$

the *intersection* refers to the individuals present in both the ground truth and the discovered group, while the *union* accounts for all individuals that appear in either group. If the Helf Ratio exceeds 0.5, the detected group is considered a *true detection*. By this, we can compute the *precision*, *recall*, and *F1 score*.

We also adopted the extended metrics proposed by [11]. Specifically, we evaluate the accuracy at varying thresholds by increasing the criteria from 0.5 to 1 in steps of 0.1, enabling us to plot the accuracy curve and compute the AUC (Area Under the Curve) score, referred to as IoU@AUC. Additionally, we compute the matrix IoU score (Mat.IoU) using the predicted and ground-truth binary human group relation matrices, where element-wise logical AND and OR operations are applied to measure the overlap between groups.

3.3 Network Implementation details

We used Incepction-v3[8] as a frozen backbone pretrained on Collective Activity Dataset [10], and utilize ROI-Align [9] with crop size of 5×5 to extracte features. The parameter λ in Eq. (7) is set to 0.3. The parameter τ in Eq. (6) is set to 0.7 times the width of the input image.

3.4 Training & Inference

In this study, all experiments were conducted on a workstation equipped with an AMD Ryzen 9 5950X CPU and NVIDIA RTX 3090 GPU. The workstation runs on the Ubuntu 18.04 LTS operating system. All networks in this research were implemented under the PyTorch framework. We trained the network using stochastic gradient descent (SGD) with a learning rate initially with 2×10^{-5} and decreasing to 1×10^{-5} at epoch 20. We trained the model for total 30 epochs.

3.5 Result & Discussion

This section presents the results of our proposed framework and discusses the findings. From Table 2, it can be observed that the **Precision**, **Recall**, and **F1 Score** are 64.4%, 82.7%, and 72.2%, respectively. These results indicate that the model performs well in minimizing false detections (high Precision) and capturing most of the true groups (high Recall). However, the significant difference between Recall and Precision suggests that while the model excels in identifying a wide range of groups, it may occasionally misclassify non-groups, leading to false positives. This trade-off highlights the need for further refinement to achieve better balance in precision and recall without compromising detection accuracy.

Table 2. Precision, Recall, and F1 Score (%)

Precision	Recall	F1 Score
64.4	82.7	72.2

Similarly, Table 3 provides insight into the spatial alignment of detected groups, with **IoU@0.5**, **IoU@AUC**, and **Mat.IoU** achieving 72.2%, 33.5%, and 72.3%, respectively. The high **IoU@0.5** and **Mat.IoU** values indicate that the model effectively aligns detected groups with ground truth annotations, demonstrating strong spatial consistency. However, the relatively lower **IoU@AUC** suggests that the model's performance varies across different thresholds, possibly due to challenges in handling edge cases or overlapping groups. This result underscores the importance of refining the model to improve its robustness in more complex scenarios, particularly under varying spatial conditions or occlusions.

Table 3. IoU Metrics (%)

IoU@0.5	IoU@AUC	Mat.IoU
72.2	33.5	72.3

To further analyze the results, we conducted a sensitivity analysis on the visual feature similarity R_e and spatial proximity $\tilde{R}_d$. Table 4 demonstrates that spatial proximity $\tilde{R}_d$ plays a more critical role in group detection performance. As the weight of spatial proximity increases, the performance improves, achieving the highest IoU@0.5 and IoU@AUC values of **72.2%** and **33.5%** when $R_e = 0.3$ and $\tilde{R}_d = 0.7$. However, when the model relies solely on spatial proximity ($R_e = 0$), the performance declines slightly, indicating that visual feature similarity R_e still contributes to learning individual relationships effectively. This suggests that while spatial information dominates, incorporating visual feature similarity helps the model achieve a more balanced understanding of group relationships.

Figure 3 visualizes the correct predictions. In Prediction

Table 4. Sensitivity analysis on visual feature similarity R_e and spatial proximity $\tilde{R}_d$. The highest scores are marked in **bold** and the second highest in underlined.

R_e	$\tilde{R}_d$	Group Detection		
		IoU@0.5	IoU@AUC	Mat.IoU
1		58.5	29.4	**74.1**
0.7	0.3	61.2	29.2	71.7
0.5	0.5	63.5	30.6	71.0
0.3	0.7	**72.2**	**33.5**	72.3
	1	70.6	31.5	69.7

1 (Figure 3b), the framework demonstrates its capability to effectively distinguish multiple groups within a complex scene, even when numerous workers are present and interacting across different locations. This highlights the model's ability to accurately separate groups in dense and dynamic environments. On the other hand, the result for Prediction 2 (Figure 3d) emphasizes the model's ability to correctly identify the true group despite challenges posed by spatial proximity. Specifically, although the two individuals in the upper-right corner appear close to each other, the model successfully leverages visual feature similarity to learn that the actual group consists of the two workers actively engaged in rebar installation. This underscores the model's robustness in learning semantic group relations based on contextual and task-related cues, rather than relying solely on physical distance.

However, Figure 4 highlights the incorrect predictions. The primary source of error in the upper-right corner stems from the intertwining and occlusion of two groups, making it difficult for the model to differentiate between them. Additionally, since the current model takes a single image as input, understanding group dynamics and movement becomes particularly challenging under such conditions. This limitation suggests that without temporal information, the model struggles to resolve ambiguities caused by overlapping worker interactions and visual obstructions. Beyond occlusion, we identified additional error sources: (1) Dynamic role changes, where workers temporarily move between groups, causing assignment inconsistencies; (2) Varying worker proximity, where workers working near each other but not collaboratively are mistakenly clustered together; and (3) Clothing and visual appearance similarities, where workers wearing identical uniforms and safety gear are misclassified into the same group due to similar feature embeddings. These findings suggest that integrating temporal tracking and additional contextual cues could enhance robustness in future iterations of the model.

4 Conclusions

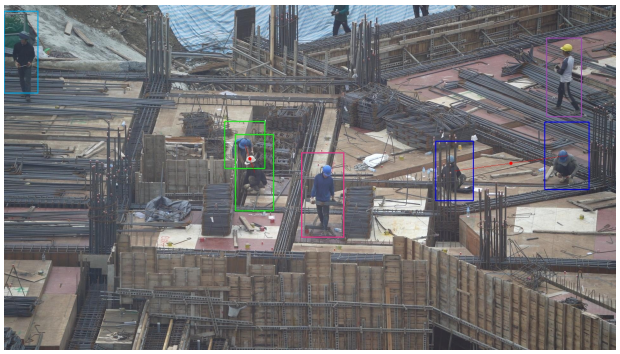
In this study, we introduced a deep learning-based framework for automated crew-level identification on construction sites. The proposed framework leverages a graph-based approach that combines visual feature similarity and spatial proximity, enabling robust detection of worker groups through clustering algorithms. By integrating these relationships into a relational matrix and applying spectral clustering, the method effectively identifies groups of workers collaborating on shared tasks.

The framework was validated on a construction site dataset collected during a rebar installation task. The experimental results showed that our method achieved high performance in group detection, with metrics such as IoU@0.5, IoU@AUC, and Mat.IoU reaching 72.2%, 33.5%, and 72.3%, respectively. Sensitivity analysis further revealed the significance of spatial proximity in improving detection accuracy, while visual feature similarity provided additional contextual support for identifying group relationships.

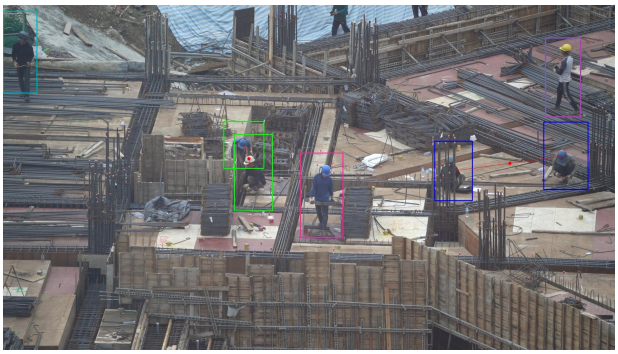
Despite the promising results, certain limitations were observed, particularly in scenarios involving heavy occlusion or overlapping worker interactions. These challenges highlight the need for incorporating temporal information to better capture group dynamics and movement across sequences of frames. While our dataset captures key worker interaction patterns and spatial relationships, we acknowledge that differences in construction site layouts, worker densities, and environmental conditions may influence the model's performance.

For practical deployment, cameras would be installed on nearby buildings or tower cranes to capture footage of the construction site. The recorded videos would then be transmitted to an office workstation, where the proposed system processes the data to detect and analyze worker groups. This setup enables real-time monitoring without interfering with ongoing construction activities. While the initial setup requires hardware installation and network configuration, the system significantly reduces manual observation efforts and improves productivity tracking, making it a practical and scalable solution for construction management.

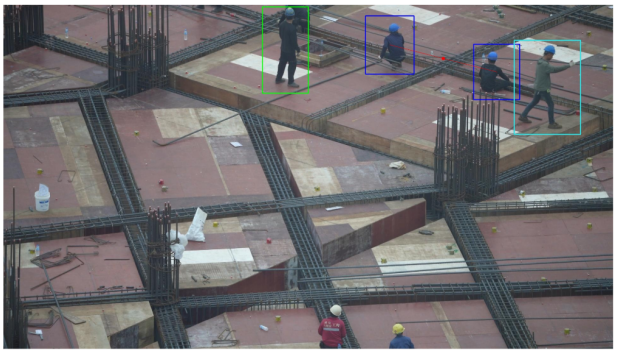
Future improvements could focus on incorporating temporal analysis to enhance worker grouping consistency across frames. One approach is to implement trajectory-based tracking algorithms to maintain worker identities and analyze movement patterns over time. This would help differentiate temporary interactions from persistent group formations. Additionally, sequential modeling techniques such as RNNs or transformer-based video models could be explored to capture temporal dependencies, further refining worker clustering in dynamic construction environments.



(a) GroundTruth 1



(b) Prediction 1



(c) GroundTruth 2

(d) Prediction 2

Figure 3. Visualization of Correct Predictions: The detected groups (b, d) match the ground truth groups (a, c).

(a) Groundtruth

(b) Prediction

Figure 4. Visualization of Incorrect Predictions: The detected groups (b) do not match the ground truth groups (a).

References

[1] Behnam Sherafat, Changbum R Ahn, Reza Akhavian, Amir H Behzadan, Mani Golparvar-Fard, Hyunsoo Kim, Yong-Cheol Lee, Abbas Rashidi, and Ehsan Rezazadeh Azar. Automated methods for activity recognition of construction workers and equipment: State-of-the-art review. *Journal of Construction Engineering and Management*, 146(6): 03120002, 2020.

[2] Jie Gong and Carlos H Caldas. An object recognition, tracking, and contextual reasoning-based video interpretation method for rapid productivity analysis of construction operations. *Automation in Construction*, 20(8):1211–1226, 2011.

[3] H Randolph Thomas and Jeffrey Daily. Crew performance measurement via activity sampling. *Journal of construction engineering and management*, 109 (3):309–320, 1983.

[4] Hanbin Luo, Chaohua Xiong, Weili Fang, Peter ED Love, Bowen Zhang, and Xi Ouyang. Convolutional neural networks: Computer vision-based workforce activity assessment in construction. *Automation in Construction*, 94:282–289, 2018.

[5] Xiaochun Luo, Heng Li, Dongping Cao, Yantao Yu, Xincong Yang, and Ting Huang. Towards efficient and objective work sampling: Recognizing workers' activities in site surveillance videos with two-stream convolutional networks. *Automation in Construction*, 94:360–370, 2018.

[6] Xiaochun Luo, Heng Li, Dongping Cao, Fei Dai, JoonOh Seo, SangHyun Lee, et al. Recognizing diverse construction activities in site images via relevance networks of construction-related objects detected by convolutional neural networks. *J. Comput. Civ. Eng*, 32(3):04018012, 2018.

[7] Jiannan Cai, Yuxi Zhang, and Hubo Cai. Two-step long short-term memory method for identifying construction activities through positional and attentional cues. *Automation in Construction*, 106: 102886, 2019.

[8] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[10] Wongun Choi, Khuram Shahid, and Silvio Savarese. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 1282–1289. IEEE, 2009.

[11] Ruize Han, Haomin Yan, Jiacheng Li, Songmiao Wang, Wei Feng, and Song Wang. Panoramic human activity recognition. In *European Conference on Computer Vision*, pages 244–261. Springer, 2022.

[12] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information processing systems*, 17, 2004.

[13] Wongun Choi, Yu-Wei Chao, Caroline Pantofaru, and Silvio Savarese. Discovering groups of people in images. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*, pages 417–433. Springer, 2014.