Fast Hand Detection in Collaborative Learning Environments*

Sravani Teeparthi[1], Venkatesh Jatla[1], Marios S. Pattichis[1], Sylvia Celedón-Pattichis[2], and Carlos LópezLeiva[2]

[1] The University of New Mexico, Albuquerque, NM, USA
{steeparthi, venkatesh369, pattichi}@unm.edu
[2] Department of Language, Literacy, and Sociocultural Studies
{sceledon, callopez}@unm.edu

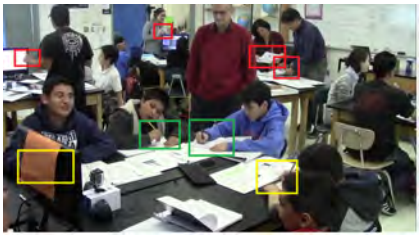
Abstract. Long-term object detection requires the integration of frame-based results over several seconds. For non-deformable objects, long-term detection is often addressed using object detection followed by video tracking. Unfortunately, tracking is inapplicable to objects that undergo dramatic changes in appearance from frame to frame. As a related example, we study hand detection over long video recordings in collaborative learning environments. More specifically, we develop long-term hand detection methods that can deal with partial occlusions and dramatic changes in appearance.
Our approach integrates object-detection, followed by time projections, clustering, and small region removal to provide effective hand detection over long videos. The hand detector achieved average precision (AP) of 72% at 0.5 intersection over union (IoU). The detection results were improved to 81% by using our optimized approach for data augmentation. The method runs at 4.7×the real-time with AP of 81% at 0.5 intersection over the union. Our method reduced the number of false-positive hand detections by 80% by improving IoU ratios from 0.2 to 0.5. The overall hand detection system runs at 4× real-time.

Keywords: Hand detection, · Video Analysis, · Data Augmentation.

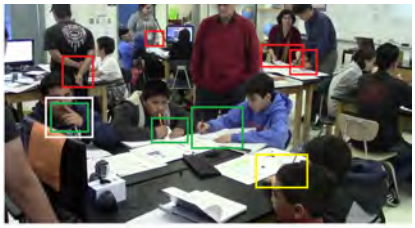
1 Introduction

We study the problem of developing a robust method for detecting student hands in collaborative learning environment [3]. Here, we define a collaborative learning environment as a small group of students working together in a single table as shown in Fig. 1. Our goal is to recognize writing and typing activities over the

* This material is based upon work supported by the National Science Foundation under the AOLME project (Grant No. 1613637), the AOLME Video Analysis project (Grant No. 1842220), and the ESTRELLA project (Grant No. 1949230). Any opinions or findings of this paper reflect the views of the authors. They do not necessarily reflect the views of NSF.



(a) Sample video frame showing fully visible hands, occluded hands, and hands belonging to other groups.



(b) Sample video frame occuring 2 seconds after the frame in (a). On the lower-left, a new set of hands appears.

Fig. 1: Hand detection in collaborative learning environments. The problem is restricted to detecting student hands that are nearer to the camera. We use green bounding boxes to identify unobstructed hands that need to be detected. We use yellow bounding boxes to identify occluded hands that need to be detected through projection methods. We use red bounding boxes to identify hands that belong to groups that are associated with hands outside our group of interest. We use a white bounding box in (b) to highlight the appearance of a hand that was fully occluded in (a).

detected hand regions. We will then use the writing and typing activities to assess student participation.

For robust detection, we require that our hand detection results are consistent throughout the video, implying that we need to deal with occlusions. Furthermore, we need to reject hands that belong to students that belong to other groups, as opposed to the collaborative group that is closer to the camera (see Fig. 1). Since our ultimate goal is to apply our methods to about 1,000 hours of digital videos, we also require that our methods are fast.

We also recognize the dynamic aspects of the hand detection problem. First, it is clear that we need to associate hands with different people and that there is a need to deal with the fact that hands can disappear from view due to occlusion (see Fig. 1). Second, we note that the same hands assume very different appearances throughout the video and that there is a need to associate their variations with a single instance.

We summarize some earlier research on the same problem in the M.Sc. thesis by C.J. Darsey [4]. In her thesis, the author studied the problem of accurate hand segmentation over a limited dataset. The dataset consisted of 15 video clips of a maximum duration of 99 seconds. While the methods were successful over a limited video dataset, it is important to note that we are dramatically extending this prior research to long-term detection of hand regions over long video segments. Thus, unlike [4], the current paper also deals with occlusion, rejecting hands outside the group, and associating hand regions with different students. We also have an earlier attempt to detect hands using deep learning in [6]. The current paper dramatically extends this prior research that was focused

on very short video datasets without considering occlusion, appearance issues, and associating hands with different people. We also note that head detection and person recognition has been studied in [14], [15], [19] and [17]. Human activity classification over cropped regions was studied in [5], [16] and [8]. In addtion, we note that speech recognition using speaker geometry is studied in [18].

The current paper uses transfer learning from deep learning methods to provide initial hand detection results. For this initial step, we tested several well-known methods. We tested Faster R-CNN [13], YOLO [11], and SSD [9]. We then decided to adopt Faster R-CNN as our baseline model due to the fact that it is more widely supported within human activity recognition systems. We then build our system by post-processing the results from Faster R-CNN. More specifically, we project the results over short video segments to address occlusion and then develop a clustering approach and small area removal to identify the students within the current collaborative group, which are not addressed by traditional hand tracking methods (e.g., [12]). Our approach yields significant improvements over the standard use of Faster R-CNN.

The rest of the paper is organized into three additional sections. We summarize the methodology in section 2. We then present results in section 3 and provide concluding remarks in section 4.

2 Methodology

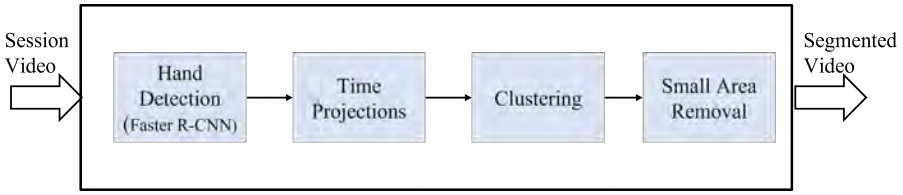
We summarize our methodology into two sections. First, we present a summary of our hand detection method. Second, we present an optimal data augmentation approach to extend our ground truth dataset.

2.1 Hand detection method

We present a block diagram and the corresponding pseudo-code of our approach in Fig. 2. We begin with a deep-learning method that detects hands at the rate of one frame per second. The output of the hand detection method is assumed to be 1 over pixel regions that represent hand regions, and 0 over other regions. Then, we take the projection of the detected regions every 12 seconds. The projected images $\{\mathtt{PI}_1, \mathtt{PI}_2, ..., \mathtt{PI}_{\lfloor \mathtt{n}/12 \rfloor}\}$ can hold a maximum of 12 that represents hand detection over all images, and a minimum of 0 that represents the lack of any hands detected over any image.

To account for occlusion, appearence, and disappearance, we apply a clustering method over the projected image. Several other standard clustering were investigated during the training process (e.g., Otsu, Li, mean, min, etc [10]). We found that ISODATA [1] performed best. ISODATA is an iterative method that uses Euclidean distance to determine the clusters.

We illustrate the proposed approach in Fig. 3. We show hand detections, obtained after non-maximum supression [2], and clusters using time projections respectively in Figs. 3a and 3b. Following this, we were able to reject out of group hand clusters with high confidence based on a cluster area constraint [7] as shown in Fig. 3d. The final clusters are then shown in Fig. 3e.



function DETECTHANDS(w_*, $\mathtt{V}$, $\mathtt{a_{th}}$)
▷ **Input**:
▷ w_* represents a pre-trained single-frame hand detector.
▷ $\mathtt{V}$ represents a short Video segment of fixed $\mathtt{n}$ seconds duration.
▷ $\mathtt{a_{th}}$ represents a minimum area requirement.
▷ **Output**:
▷ $\mathtt{H}$ contains the detected hand regions for each 12-second video segment.
$\mathtt{BI} \leftarrow w_*(\mathtt{V})$ ▷ detect hands at the rate of one frame per second.
$\mathtt{H} \leftarrow \{\}$ ▷ initialize $\mathtt{H}$ to store hand detections.
for each 12-second video segment i: **do**
Project the detected hand regions using:
$\mathtt{PI_i} \leftarrow \sum_s \mathtt{BI_s}$
Cluster the projected hand regions using:
$\mathtt{CI_i} \leftarrow$ **Cluster**$(\mathtt{PI_i})$
Remove small hand regions of far-away groups:
$\mathtt{H_i} \leftarrow$ **AreaThreshold**$(\mathtt{CI_i}, \mathtt{a_{th}})$
$\mathtt{H} \leftarrow$ **Append**$(\mathtt{H}, \mathtt{H_i})$
end for
return $\mathtt{H}$
end function

Fig. 2: Proposed hand detection method using time-projections, clustering, and small region removal

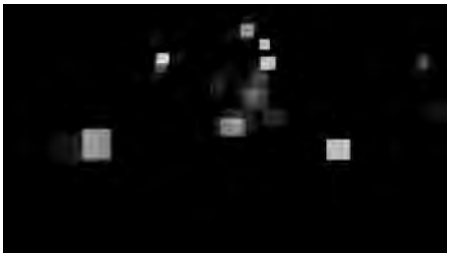
2.2 Optimal data augmentation

For robust detection, developed an optimization method for augmenting the dataset. Our goal here is to significantly extend the hand dataset for different scenarios.

The hand detection dataset was created by extracting frames from 44 different collaborative learning sessions. These sessions were selected across 3 years providing a diverse dataset. We labeled every hand instance for a total of $4,548$ instances. We partition the dataset into training, validation, and testing samples as given in Table 1.

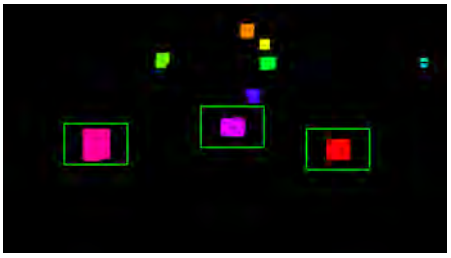
The ground truth images span multiple video sessions. For training, we sampled hands from 33 video sessions. For validation, we sampled hands from another four video sessions. For testing, we used another set of 7 complete video sessions. Video sessions were collected over three years. Video sessions were forty-five to one hour and fifteen minutes long. The training dataset described in Table 1 was carefully selected to have diversity with 350 samples.

(a) Hand detections using Faster R-CNN.



(b) Hand detection projections for 12 seconds.

(c) Binary image showing clusters



(d) Green boxes showing valid clusters after removing small clusters.

(e) Hand detections using our method.

Fig. 3: Hand detection images that demonstrate the proposed approach.

Table 1: Dataset for training, validation, and testing. The training, validation, and testing examples come from different video sessions.

	# Sessions	**# Images**	**# hand instances**
Training	33	305	1803
Validation	4	100	714
Testing	7	313	2031
Total	44	718	4,548

We develop a separable optimization approach that starts with determining the maximum range of angles for shear, rotation, and pixels to be translated. To establish the maximum range of values to consider for shear and rotation, we calculate validation accuracy at multiple angles: $\theta \in \{1°, 2°, 4°, 8°, 16°, 32°\}$. The

Table 2: Optimal augmentation parameter value ranges.

Method	Optimal range
Shear	$[-3°, 3°]$
Rotate	$[-7°, 7°]$
Translate	$[-20, 20]$

maximum range is determined based on the largest angle that results in a significant decrease in validation accuracy. Let [-θ_r^*, θ_r^*], [-θ_s^*, θ_s^*] denote the optimal ranges for rotation and shear, respectively. Similarly, we evaluate validation accuracy at multiple horizontal translations: $\tau \in \{1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 800\}$, and compute the maximum interval: $[-\tau^*, \tau^*]$. We summarized augmentation methods, along with their respective optimal ranges in Table 2.

In addition to determining the best parameter values for each augmentation method, we also optimize the probability, $\mathtt{p}$, for applying data augmentation. For example, for $\mathtt{p} = 1$, data augmentation is always applied. We compute the optimal data augmentation probability $\mathtt{p}^*$ as described in Fig. 4.

3 Results

We present the results in two sections. We first present improvement in hand detection by using optimal data augmentation method described in section 2.2. We then present the final detetion results that demonstrate that our method reduced the number of false positive regions by 78.8% without sacrificing any true positive detections.

We used an Intel Xeon 4208 CPU @ 2.10GHz server, having 128 GB DDR4 RAM and an NVIDIA RTX 5000 GPU for all the experiments. For training Faster R-CNN, we used the recommended learning rate of 0.001 for 12 epochs

for each $\mathtt{p} \in \{0, 0.25, 0.5, 0.75, 1\}$ **do**
  for each image in training **do**
    Apply **random horizontal flips** with $\mathtt{p}$ probability.
    Apply **random scaling** of {0.8,1.2} with $\mathtt{p}$ probability.
    Apply **random shear angle** sampled from $\{-\theta_s^*, ..., \theta_s^*\}$ with $\mathtt{p}$ probability.
    Apply **random rotation angle** sampled from $\{-\theta_r^*, ..., \theta_r^*\}$
      with prbability $\mathtt{p}$.
    Apply **random horizontal translation** with pixels
      uniformly sampled from $\{-\tau^*, ..., \tau^*\}$ with $\mathtt{p}$ probability.
  end for
  Train the model with the augmented data.
  Record validation accuracies.
end for
Select optimal probability ($\mathtt{p}^*$) that has the highest validation accuracy

Fig. 4: Pseudocode for finding the optimal probability for data augmentation.

Table 3: Hand detection validation and testing average precision. From the table, it is clear that `p` of 0.5 gave the best performance.

Data split	**Model**	**Probability of applying each data augmentaion**				
		0	**0.25**	**0.5**	**0.75**	**1.0**
Val	Best	0.77	0.86	**0.86**	0.85	0.84
	Last	0.76	0.85	**0.86**	0.84	0.82
Test	Best	0.75	0.80	**0.80**	0.79	0.78
	Last	0.71	0.80	**0.81**	0.78	0.76

Table 4: Reduction in number of hand detections for each test session.

Session	**# Hand detections**		**Median IoU**		
	Faster RCNN	**Ours**	**Faster RCNN**	**Ours**	**Reduction**
C1L1P-C, Mar30	55,914	9,804	0.22	0.38	82.5%
C1L1P-C, Apr13	34,665	8,028	0.18	0.45	76.8%
C1L1P-E, Mar02	50,312	9,968	0.15	0.46	80.0%
C2L1P-B, Feb23	48,073	9,924	0.22	0.47	79.3%
C2L1P-D, Mar08	31,875	7,724	0.27	0.40	75.7%
C3L1P-C, Apr11	36,757	9,536	0.23	0.43	74.0%
C3L1P-D, Mar19	57,319	9,536	0.23	0.54	83.3%

with a mini-batch size of 2 images. We can train the model in less than 13 minutes.

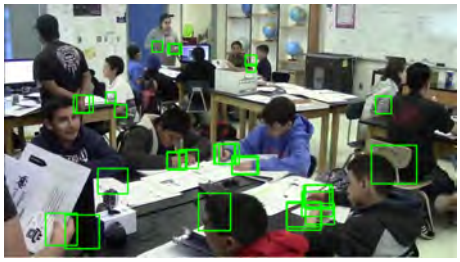
3.1 Results for optimal data augmentation

Table 2 provides the optimal maximum range angles for shear, rotation, and pixels to be translated for hand detection. We applied the optimal augmentation values at different probabilities as summarized in table 3. From this table, it is clear that 0.5 probability provided the best performance.

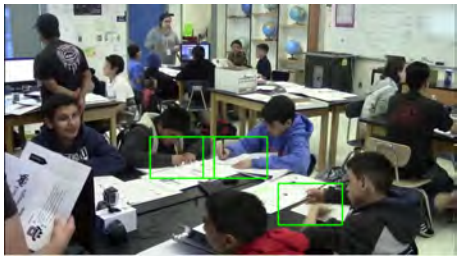
3.2 Hand detection results

We summarize our results in table 4. Compared to Faster R-CNN, our approach reduced the number of false positives by 80% while improving IoU ratios from 0.2 to 0.5. Overall, our hand detector achieved average precision (AP) of 72% at 0.5 intersection over union (IoU). The detection results were improved to 81% by using our optimized approach for data augmentation. Our method runs at 4.7×the real-time.

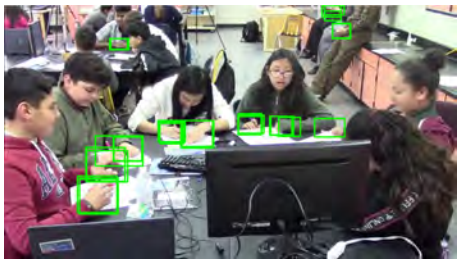
We present results against Faster R-CNN in Fig. 5. Overall, we can see that our approach results in a significant reduction in the number of detected hand regions. In some instances, our approach produces two overlapping hand regions that are associated with the same student.



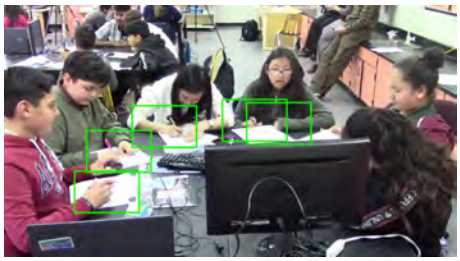
(a) Initial hand regions detected using Faster RCNN.



(b) Ours.

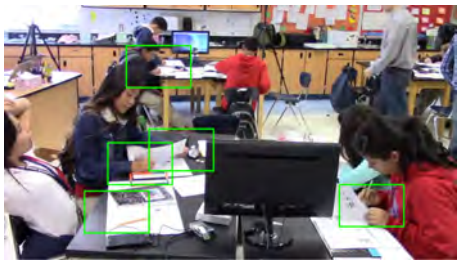


(c) Initial hand regions detected using Faster RCNN.



(d) Ours.

(e) Initial hand regions detected using Faster RCNN with significant hand movements.



(f) Ours.

Fig. 5: Comparison between Faster RCNN (left column) and our proposed approach (right column).

4 Conclusion

We presented a fast and robust method for detecting hands in collaborative learning environments. Our method performed significantly better than the standard use of Faster R-CNN. In future work, the detected proposal regions will be used for the accurate detection of writing and typing activities which can inform educational researchers identify moments of interest in collaborative learning environments.

References

1. Ball, G.H., Hall, D.J.: Isodata, a novel method of data analysis and pattern classification. Tech. rep., Stanford research inst Menlo Park CA (1965)
2. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-nms – improving object detection with one line of code. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
3. Celedón-Pattichis, S., LópezLeiva, C.A., Pattichis, M.S., Llamocca, D.: An interdisciplinary collaboration between computer engineering and mathematics/bilingual education to develop a curriculum for underrepresented middle school students. Cultural Studies of Science Education **8**(4), 873–887 (2013)
4. Darsey, C.J.: Hand movement detection in collaborative learning environment videos (2018)
5. Eilar, C.W., Jatla, V., Pattichis, M.S., LópezLeiva, C., Celedón-Pattichis, S.: Distributed video analysis for the advancing out of school learning in mathematics and engineering project. In: 2016 50th Asilomar Conference on Signals, Systems and Computers. pp. 571–575. IEEE (2016)
6. Jacoby, A.R., Pattichis, M.S., Celedón-Pattichis, S., LópezLeiva, C.: Context-sensitive human activity classification in collaborative learning environments. In: 2018 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI). pp. 1–4. IEEE (2018)
7. Jatla, V., Pattichis, M.S., Arge, C.N.: Image processing methods for coronal hole segmentation, matching, and map classification. IEEE Transactions on Image Processing **29**, 1641–1653 (2019)
8. Jatla, V., Teeparthi, S., Pattichis, M.S., Celedón-Pattichis, S., Leiva, C.L.: Long-term human video activity quantification of student participation. In: 2021 55th Asilomar Conference on Signals, Systems, and Computers. IEEE (2021)
9. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
11. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
12. Rehg, J.M., Kanade, T.: Visual tracking of high dof articulated structures: An application to human hand tracking. In: Eklundh, J.O. (ed.) Computer Vision — ECCV '94. pp. 35–46. Springer Berlin Heidelberg, Berlin, Heidelberg (1994)
13. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence **39**(6), 1137–1149 (2016)
14. Shi, W., Pattichis, M.S., Celedón-Pattichis, S., LópezLeiva, C.: Dynamic group interactions in collaborative learning videos. In: 2018 52nd Asilomar Conference on Signals, Systems, and Computers. pp. 1528–1531 (Oct 2018)
15. Shi, W., Pattichis, M.S., Celedón-Pattichis, S., LópezLeiva, C.: Robust head detection in collaborative learning environments using am-fm representations. In: 2018 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI). pp. 1–4 (April 2018). https://doi.org/10.1109/SSIAI.2018.8470355

16. Shi, W., Pattichis, M.S., Celedón-Pattichis, S., Leiva, C.L.: Person detection in collaborative group learning environments using multiple representations. In: 2021 55th Asilomar Conference on Signals, Systems, and Computers. IEEE (2021)
17. Shi, W., Pattichis, M.S., Celedón-Pattichis, S., Leiva, C.L.: Talking detection in collaborative learning environments. In: 19th International Conference CAIP. Springer (2021)
18. Tapia, L.S., Pattichis, M.S., Celedón-Pattichis, S., Leiva, C.L.: Bilingual speech recognition by estimating speaker geometry from video data. In: 19th International Conference CAIP. Springer (2021)
19. Tran, P., Pattichis, M.S., Celedón-Pattichis, S., Leiva, C.L.: Facial recognition in collaborative learning videos. In: 19th International Conference CAIP. Springer (2021)