Summer 7-28-2018

Hand Movement Detection in Collaborative Learning Environment Videos

Callie J. Darsey
University of New Mexico

Callie Jean Darsey

Candidate

Electrical and Computer Engineering

Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

Prof. Marios Pattichis , Chairperson

Prof. Balasubramaniam Santhanam

Prof. Mark Gilmore

Hand Movement Detection in Collaborative Learning Environment Videos

by

Callie Jean Darsey

B.S., Electrical Engineering, University of New Mexico, 2015

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Computer Engineering

The University of New Mexico

Albuquerque, New Mexico

July, 2018

Dedication

To the Lord, to my parents, Gary Darsey and Penny Darsey, and to Marios Pattichis. Thank you for your grace towards me.

"And further, by these, my son, be admonished: of making many books there is no end; and much study is a weariness of the flesh." Ecclesiastes 12:12

Acknowledgments

I would like to thank my advisor, Dr. Marios Pattichis, first for his advisement during this thesis and second for all the time he has spent teaching me during the last six years in which we have been in acquaintance. I would also like to thank Venkatesh Jatla, Wenjing Shi, and Abby Jacoby for handling my "may I have" and "how" questions; your theses inspired me. Lastly, I would to say hi to lab mates, especially Gangadharan Esakki who shares chocolate.

Hand Movement Detection in Collaborative Learning Environment Videos

by

Callie Jean Darsey

B.S., Electrical Engineering, University of New Mexico, 2015

M.S., Computer Engineering, University of New Mexico, 2018

Abstract

Human activity detection in digital videos is currently attracting significant research interest. This problem is especially challenging for video datasets that have a lot of human activity, illumination noise, and structural noise. The video dataset associated with the Advancing Out of School Learning in Mathematics and Engineering (AOLME) project has these challenges. ALOME videos have been used in the study of human activities "in the wild".

This thesis explores detection of hand movement using color and optical flow. Exploratory analysis considered the problem component wise on components created from thresholds applied to motion and color. The proposed approach uses patch color classification, space-time patches of video, and histogram of optical flow. The approach was validated on video patches extracted from 15 AOLME video clips. The approach achieved an average accuracy of 84% and an average receiver operating characteristic area under curve (ROC AUC) of 89%.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

The AOLME dataset has many examples of human activities and actions to analyze, (see Fig. 1.1). As the frames in Fig. 1.1 depict, in the AOLME project, students interact with facilitators and each other to work on math and engineering lessons. The videos in Fig. 1.1 have much in common. Each group has a primary camera capturing it. Each group has monitors. Each group has one keyboard which is passed around between students. There are two common room locations; thus, there are common colors (e.g. the table tops and the chairs and the cabinets). There are common students and facilitators in different videos.

However, there is a lot of variables in the videos that qualify the AOLME dataset as an uncontrolled environment. Camera angles differ, even between videos on the same group, (e.g. see Fig. 1.1f and Fig. 1.1l). The illumination is not consistent, (e.g. see Fig. 1.1b versus Fig. 1.1d). People can and do move around freely.

From the point of view of using optical flow as motion information to detect

human actions, there are also several challenges. First, there is illumination noise. Second, there are obstructions of motions. The monitors often block or partially block the view of people's movement; see how the monitors block people's hands in Fig. 1.1d, Fig. 1.1e, Fig. 1.1l, and Fig. 1.1n. People also block other people. See in Fig. 1.1e how there are multiple people stacked on the left. See in Fig. 1.1n how one girl blocks another girl on each side of the table. People block themselves. For example, a person may put a hand over their mouth which bocks an activity of talking, or for example, a person's head may block part of her moving hand, as is happening in Fig. 1.1b. Third, there is movement not associated with the primary table in a shot. Other groups work and move in the background; see Fig. 1.1a - Fig. 1.1c, Fig. 1.1e - Fig. 1.1f, Fig. 1.1h – Fig. 1.1m, and Fig. 1.1o. People walk in the background; see Fig. 1.1c, Fig. 1.1f, Fig. 1.1h - Fig. 1.1j, and Fig. 1.1l. People adjust equipment; see Fig. 1.1k and Fig. 1.1m. People hold animated conversations nearby; see Fig. 1.1n. People walk between the camera and the primary table; see Fig. 1.1l. Fourth, the people who interact at the primary table may not be sitting at the table. Participants at a primary table may stand or lean near the table; see Fig. 1.1d - Fig. 1.1f and Fig. 1.1m. Furthermore, participants may join a table partway or just visit a primary table or move to and fro from the table; see Fig. 1.1g and Fig. 1.1o and Fig. 1.1d.

1.2 Motivation

A focus of the AOLME project is understanding how the students best learn. Therefore, how the students interact with the facilitator, each other, and their lessons is of interest when defining what human activity and actions are useful to detect. What the participants are doing with their hands is an important aspect of how they interact with their lessons and each other. For example, the participants interact with

the lessons by writing (Fig. 1.1a - Fig. 1.1b, by typing (Fig. 1.1d, Fig. 1.1g, Fig. 1.1k, Fig. 1.1l, Fig. 1.1m, Fig. 1.1o), by using the mouse (Fig. 1.1o), or by flipping through their notebooks (Fig. 1.1k). They interact with each other by pointing (Fig. 1.1a, Fig. 1.1c, Fig. 1.1d, Fig. 1.1e, Fig. 1.1h, Fig. 1.1i, Fig. 1.1j, Fig. 1.1n). They interact by gesturing while talking (Fig. 1.1b). They use gestures to describe (Fig. 1.1f, Fig. 1.1i). They use gestures to communicate (in Fig. 1.1l the facilitator is giving thumbs up).

Thus the focus of this thesis is detecting moving hands.

1.3 Thesis Statement

The thesis of this research is that hand movement can be detected in full frame video based on motion information from optical flow over a duration of time and color information. For the color information, this thesis claims that skin regions can be defined by classification of patches in a video frame. The basic idea for the hand movement detection is to look everywhere in the video via space-time patches and reject regions that do not qualify due to low motion and low skin region presence. Non-rejected space-time patches can be classified based on histograms of the flow.

1.4 Contributions

The contributions of this thesis include:

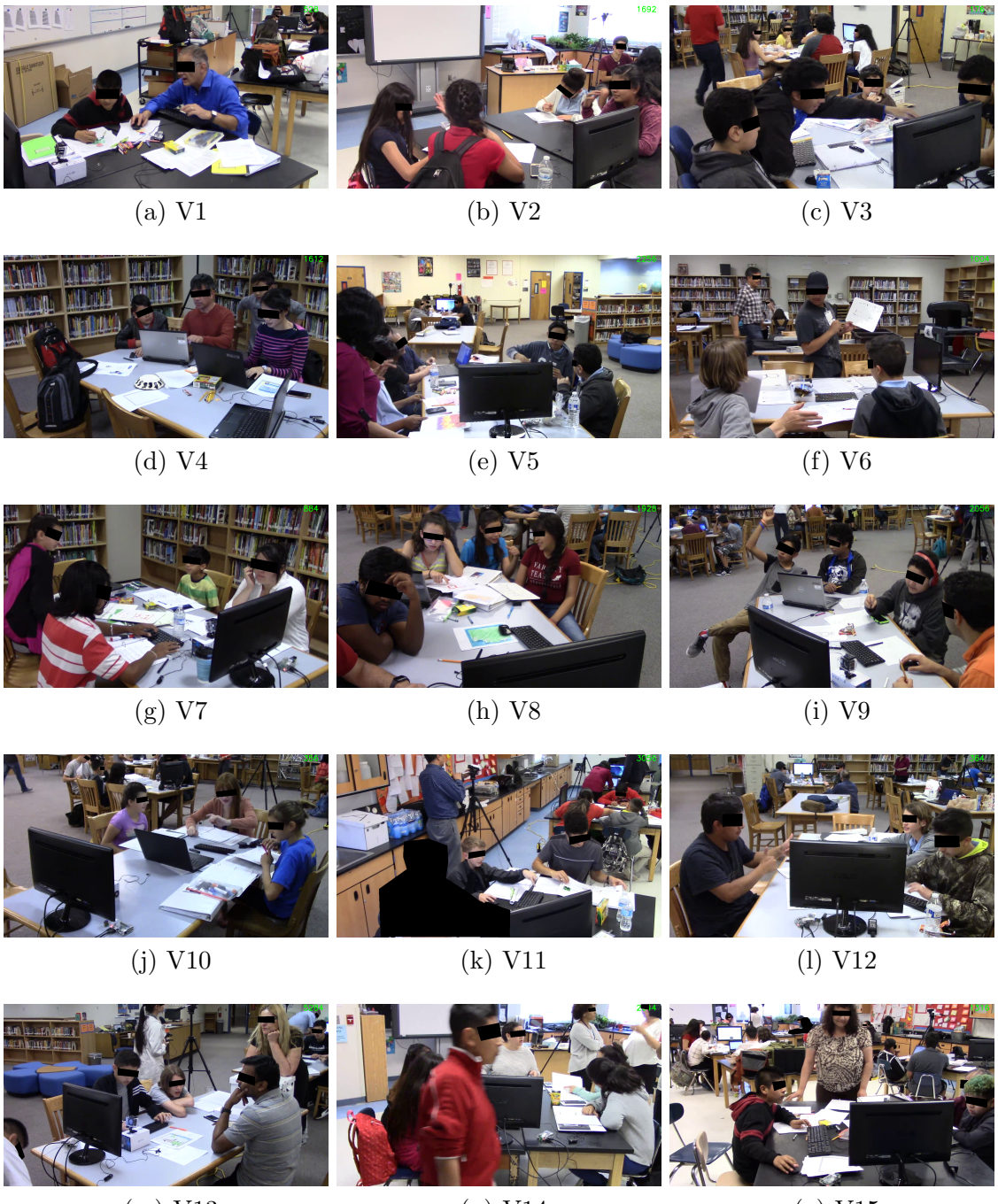
1. Exploratory analysis which considers the moving hand detection problem component wise on components created from thresholds applied to motion and color.

2. A new method for determining color regions based on patch classification.

3. A hand detection method for space-time patches based on color regions and optical flow.

1.5 Summary

Chapter 2 gives background, especially concerning research done in the AOLME dataset. Chapter 3 describes the steps used in exploratory analysis and in the space-time patch classification. Chapter 4 presents results, and Chapter 5 summarizes the conclusions and conjectures about future work.



(a) V1 (b) V2 (c) V3

(d) V4 (e) V5 (f) V6

(g) V7 (h) V8 (i) V9

(j) V10 (k) V11 (l) V12

(m) V13 (n) V14 (o) V15

Figure 1.1: Frames from 15 different videos in the AOLME dataset. Censoring is present to protect subject privacy.

Chapter 2

Background

2.1 Prior Work Done Using AOLME Dataset

References [5][4] used typing / no typing and writing / no writing video crops from AOLME videos to demonstrate a distributed and scalable video analysis architecture. Unlike [5][4], this thesis works in the full uncropped video.

References [12][13] focus on face and head detection, attention detection based on where the faces look, and group interactions based on the attention direction detected. In [12][13], the work uses texture by using an AM-FM images. In [12], the method for face detection included a pixel value based skin detector from [2]. This thesis used that skin detector during the exploratory analysis phase. Furthermore in [12], the full image was scanned in a space patch manner classifying the patches as face or non-face. Similarly, this thesis scans using patches for the hand and non-hand; however, since the search is in segments of video, space-time patches are used.

The references [9][8] focus on activity detection of writing, typing, and talking. The approach of [9][8] first finds candidate regions for writing, typing, and talking.

The candidate region for writing is found via combined pixel color value masks for tables, pencils and pens, and paper along with some shape info on the pencils. The candidate region for typing is found via pixel color value masks for keyboards and for table (with convex hull applied) and for skin along with a KNN to detect keyboards. The candidate region for talking is found via the pixel color value masks for skin, skin cleanup, KNN for faces, and shape. Once the candidate regions, which already have a context, are established classification on optical flow histograms over three frames is done to determine writing versus no writing, typing versus no typing, and talking versus no talking.

This thesis is similar to [9][8] in that both skin detection and motion information through time is used. However, this thesis approaches differently with space-time patches because general hand movement does not have a context with other objects like typing and writing. This thesis also collects features from flow over a longer duration of time than [9][8]. Lastly, while [9][8] pixel color mask for skin was considered, it was not used in this thesis's method.

2.2 Other Work Concerning Hand Detection

Examples of research that deals with hand detection are summarized in Table 2.1, and some common datasets are summarized in Table 2.2. Some of the studies had pieces that were similar to this thesis along with dissimilar pieces. The study in [7] has a patch based skin detector for RGB images. It classifies via features passed to a random forest. Unlike their study, this thesis does not extract features but directly classifies the RGB values. Furthermore, their videos used are focused on the hand (i.e. no faces are present in the videos). The study in [11] uses a convolutional neural net (CNN) for a skin detector on 5x5 overlapping patches. Unlike their study, this thesis did not use overlapping patches and used less complex classifiers. The study

in [11] also rejected regions with low skin count. This thesis does similarly; however, this thesis looks for moving hands whereas the study looks at images.

Study	Dataset	Problem Type
AHD: Thermal Image-Based Adaptive Hand Detection for Enhanced Tracking System, **2018** [14]	their own thermal camera database, palm facing videos of tracing 0-9 and A-J	detection and tracking of hand
Towards transferring grasping from human to robot with RGBD hand detection, **2017** [7]	their own generated RGBD 325 frame videos of just hand in various challenging lighting, background color, and occlusion situations	hand detection for the purposes of passing information to robot
Deep Learning Based Hand Detection in Cluttered Environment Using Skin Segmentation, **2017** [11]	public datasets of Oxford, 5-signer and EgoHands; extracted images from Indian classical dance (ICD) videos	hand detection

Table 2.1: This table summarizes types of research problems involving hand detection in current literature.

Dataset Name	Dataset Description
Oxford [10]	13050 annotated hands in images from public sources
5-signer [3][11]	footage of signers against moving backgrounds
EgoHands [1]	48 first-person videos with 15000 hand instances.

Table 2.2: This table summarizes some public datasets referenced in recent literature.

Chapter 3

Methods

3.1 Overview

This thesis was a mix of exploratory stages where comprehensive quantitative results were not the objective and of a stage where quantitative validation was recorded for an approach. The following sections are presented in the order they were considered in during research so that we build up to the final approach.

In Section 3.2, we describe a choice made after visual inspection to sample the video before making the optical flow calculations. This choice is used in subsequent exploration and approach. In Section 3.3, an exploratory stage is described where we classified components formed per frame. During this exploration, we became interested in using a skin region detection. Section 3.4, explains the motivation for and the new proposed approach for a patch skin region detector. The patch skin region detector is later used for the final approach that is quantitatively validated.

Next, we considered how to approach the hand detection over a duration of time. Section 3.5 describes how the reference dataset of space-time exemplars is

formed. In Section 3.6, we describe another component exploration stage, this time on components projected through time. Then the final space-time patch approach is described in Section 3.7.

A description of the computing platform that was used for implementing the final system is given in Section 3.8.

3.2 Sampling and Resizing Video before Optical Flow Calculation

Motion information is obtained using optical flow [6]. To cause humans motion to be more separable from small illumination movement in the background, we calculate optical flow on frames that are four apart. The videos we used are 60 frames per second. See Fig. 3.1 for an example of the distinction. We also resized the 1920 x 1080 images to 724 x 1286 for faster processing, easier viewing, and magnifying the motions.

(a) zero frames dropped (b) three frames dropped

Figure 3.1: Optical flow for different number of frames dropped. Censoring is present to protect subject privacy.

3.3 Classification of Per Frame Components

In this section, the method and results from exploration of per frame components is presented.

3.3.1 Method of Classification of Per Frame Components

The basic idea was to capture moving hands in a frame by intersecting skin regions [12][2] (Fig. 3.4) with flow magnitudes (Fig. 3.3a or Fig. 3.3b). See Fig. 3.5 for intersection. Classification of the components as *hand* or *non-hand* would be done from features associated with the component. The flow threshold was frame dependent.

Due to how the skin regions captured tended to be on the edge of hands Fig. 3.4, further processing steps were needed. The approach is shown in Fig. 3.6, Fig. 3.7, and Fig. 3.8 and described in Fig. 3.9. We will refer to this procedure as `approach_1`.

To break up small connections between components, slight variations on the `approach_1` components via with watershed lines were added into `approach_2` and `approach_3`. These are depicted in Fig. 3.10, Fig. 3.11, Fig. 3.12, Fig. 3.13, Fig. 3.14, Fig. 3.15, Fig. 3.16, Fig. 3.17, and Fig. 3.19 and described in Fig. 3.18 and Fig. 3.20.

Labeling of the components for training was done via using exemplar boxes that we created as reference. See Fig. 3.21. The exemplar boxes were drawn tightly around all the hands in a frame. If 20% of a component is overlapped by region that fall in exemplar boxes, then the component is considered a *hand* component. Otherwise it is labeled as a *non-hand* component. See Fig. 3.22, Fig. 3.23, and Fig. 3.24.

The features extracted for each component are as follows:

- Sixteen bin hue probability density function (PDF).

- Sixteen bin saturation PDF.
- Eighteen bin value PDF.
- Canny edge to component area ratio.
- Eight bin histogram of flow with each bin normalized by the number of values falling the bin.
- Component area.
- Component perimeter.
- Orientation of minimum area rectangle.
- Ratio of component area to minimum area rectangle area.
- Ratio of minimum area rectangle's short side to its long side.

For classification, random forest classifier is trained with bagging enabled. Therefore the out-of-bag (OOB) samples are available for validation testing. See Section 3.3.2 for those results.

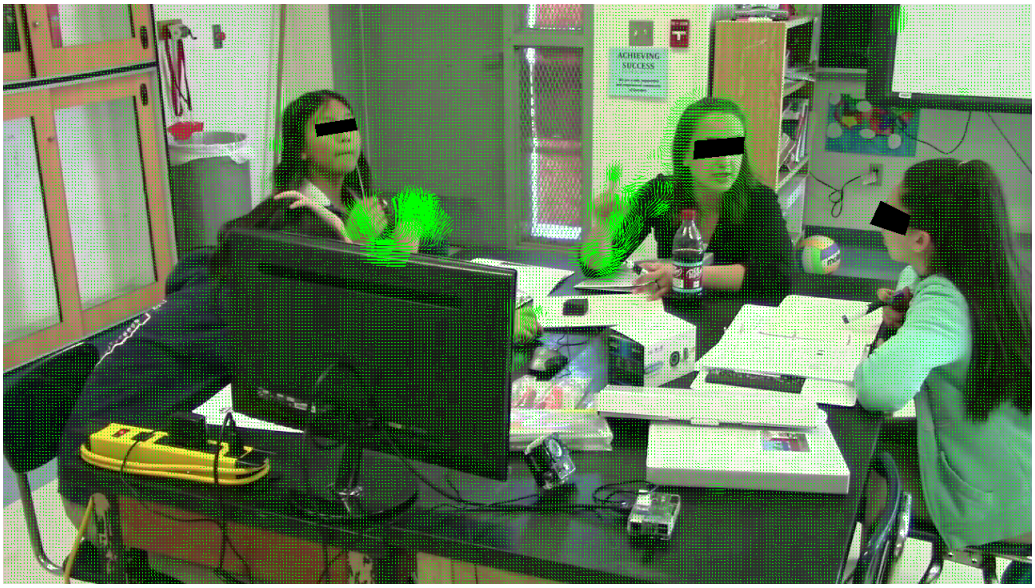


Figure 3.2: `flow` step of Fig. 3.9. Censoring is present to protect subject privacy.

(a) `flow_thresh`

(b) `flow_thresh (different visualization)`

Figure 3.3: `flow_thresh` step of Fig. 3.9. Censoring is present to protect subject privacy.

Figure 3.4: `skin_detection`[12][2] step of Fig. 3.9.

Figure 3.5: `intersection` step of Fig. 3.9.

Figure 3.6: `dt_intersection` step of Fig. 3.9.

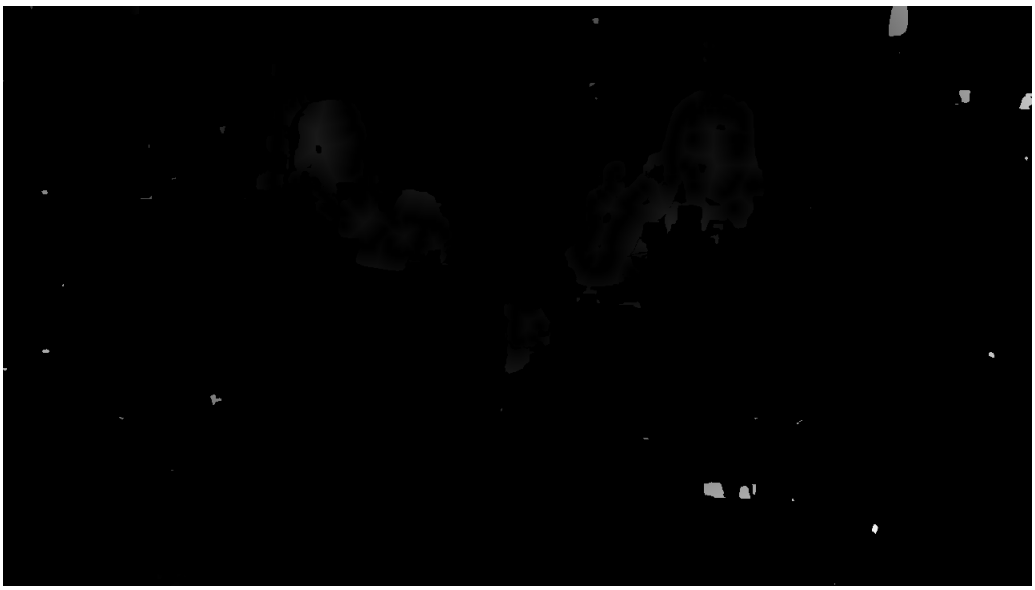


Figure 3.7: `importance` step of Fig. 3.9.

Figure 3.8: `approach_1_cpns` step of Fig. 3.9. Censoring is present to protect subject privacy.

---

function APPROACH_1_COMPONENTS(`frame`, `flow_mag`)

`flow_thresh` ← `flow_mag` > MEDIAN of `flow_mag` where
`flow_mag` > Q3(`flow_mag`) +
1.5 * IQR(`flow_mag`)
`skin_detection` ← SKIN_DETECTOR[12][2](`frame`)
`intersection` ← LOGICAL_AND(`flow_thresh`, `skin_detection`)
`dt_intersection` ← DISTANCE_TRANSFORM(`intersection`)
`importance` ← `dt_intersection` where `flow_thresh` exist

▷ approach 1 components
`approach_1_cpns` ← `importance` > 14.5

return `approach_1_cpns`
end function

Figure 3.9: Steps taken to get the components (we refer to this as approach_1).

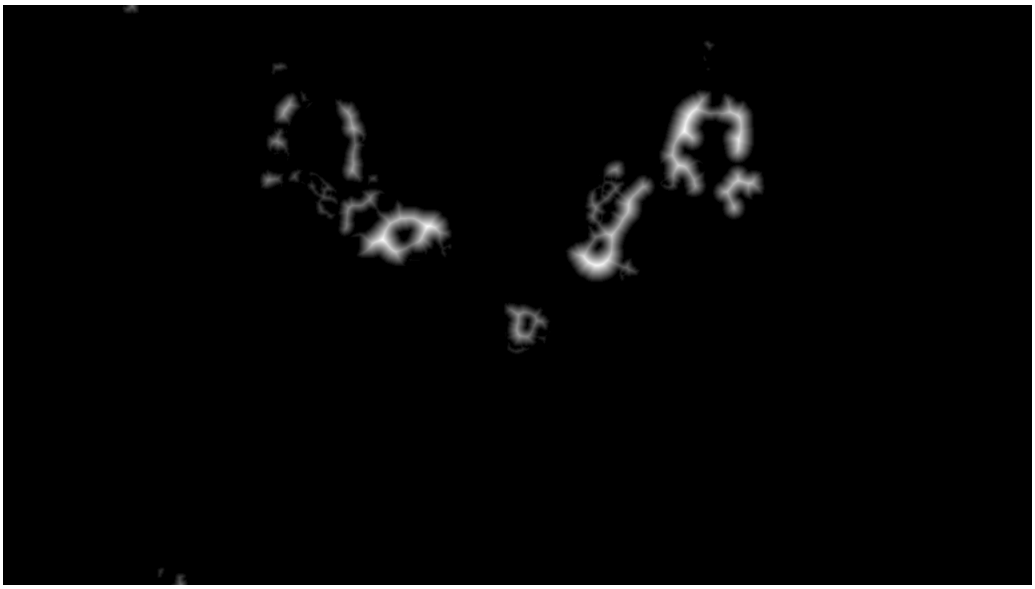


Figure 3.10: `dt_approach_1` step of Fig. 3.18.

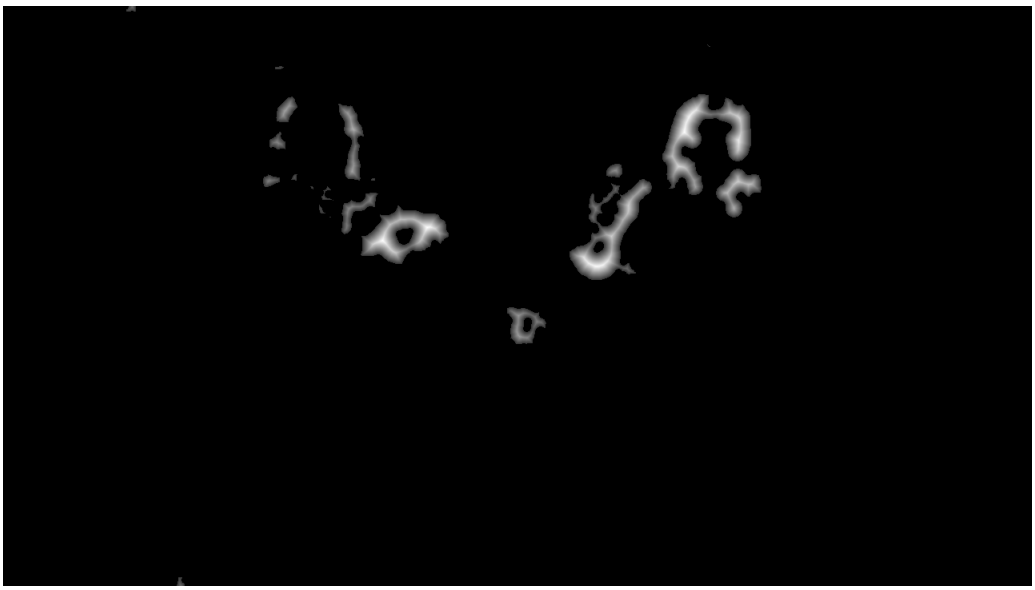


Figure 3.11: `dt_approach_1_less_connectors` step of Fig. 3.18.

Figure 3.12: `top_75_percent` step of Fig. 3.18.

Figure 3.13: `local_max` step of Fig. 3.18.

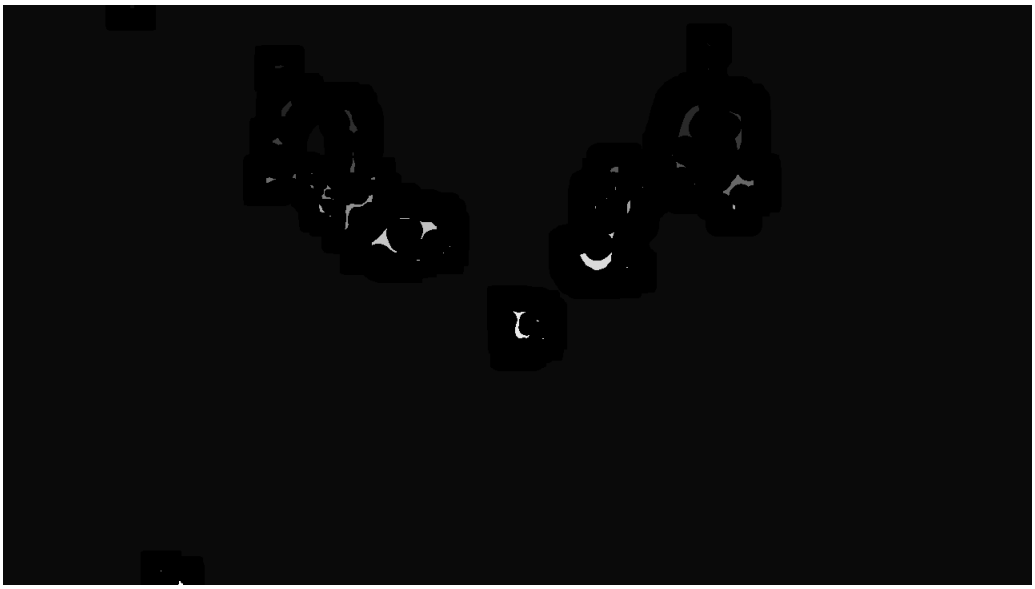


Figure 3.14: `markers` step of Fig. 3.18.

Figure 3.15: `WS_result` step of Fig. 3.18.

(a)

Figure 3.16: `WS_cpns` step of Fig. 3.18 (watershed components)

Figure 3.17: `approach_2_cpns` step of Fig. 3.18 (approach 2 components)

---

function APPROACH_2_COMPONENTS(`approach_1_cpns`, `frame`)

`dt_approach_1` ← DISTANCE_TRANSFORM(`approach_1_cpns`)
`dt_approach_1_less_connectors` ← **remove** regions less than
0.2 * MAX(`dt_approach_1`)
`top_75_percent` ← **take** the top 75% per component in
`dt_approach_1_less_connectors`
`local_max` ← **mark** pixel if it is the max value in a 13 x 13 neighborhood
`markers` ← **get** the unknown region in the markers as dilation around
LOGICAL_OR(`top_75_percent`, `local_max`)
`WS_result` ← WATERSHED(`frame`, `markers`)

▷ watershed components
`WS_cpns` ← **enlarge** watershed lines in `WS_result`
`approach_2_cpns` ← **break up** the `approach_1_cpns` with lines
between `WS_cpns`

return `approach_2_cpns`, `WS_cpns`
end function

Figure 3.18: Additional steps to get variation from approach_1 of Fig. 3.9

Figure 3.19: `approach_3_cpns` step of Fig. 3.20 (approach 3 components)

---

function APPROACH_3_COMPONENTS(`WS_cpns`, `approach_1_cpns`)

▷ approach 3 components
`approach_3_cpns` ← **take** `WS_cpns` that overlap
`approach_1_cpns`
return `approach_3_cpns`
end function

Figure 3.20: Additional steps to get variation from approach_1 of Fig. 3.9 using `WS_cpns` generated in Fig. 3.18

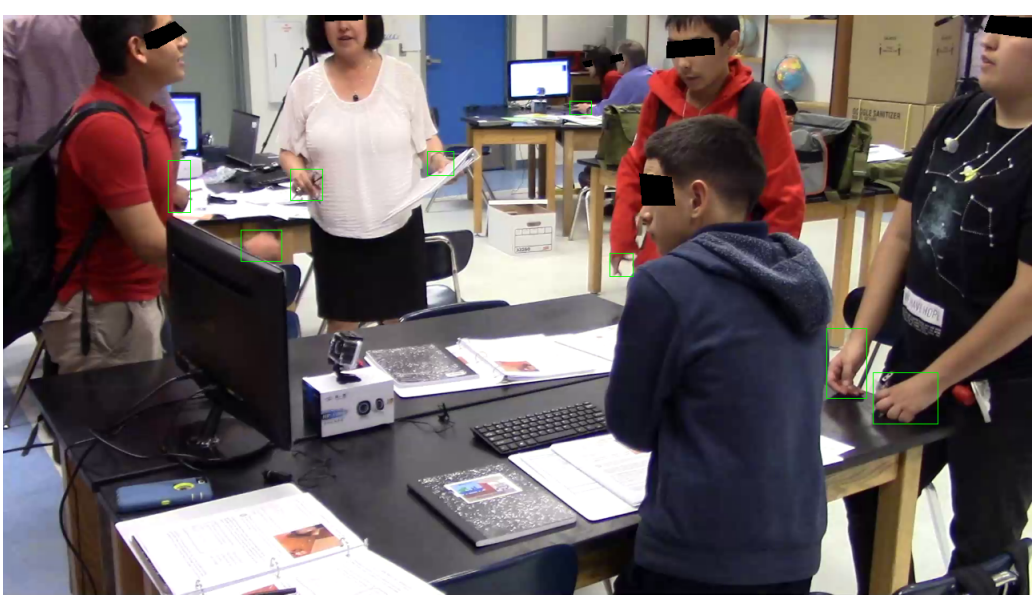


Figure 3.21: The exemplar boxes are drawn closely around all hands in the frame.

Figure 3.22: This figure shows approach_1 labels. The green components are labeled *hand*, and the red components are labeled *non-hand* based on how the component overlaps exemplar regions of Fig 3.21. Censoring is present to protect subject privacy.

Figure 3.23: This figure shows approach_2 labels. The green components are labeled *hand*, and the red components are labeled *non-hand* based on how the component overlaps exemplar regions of Fig 3.21. Censoring is present to protect subject privacy.

Figure 3.24: This figure shows approach_3 labels. The green components are labeled *hand*, and the red components are labeled *non-hand* based on how the component overlaps exemplar regions of Fig 3.21. Censoring is present to protect subject privacy.

3.3.2 Component Classifier Results

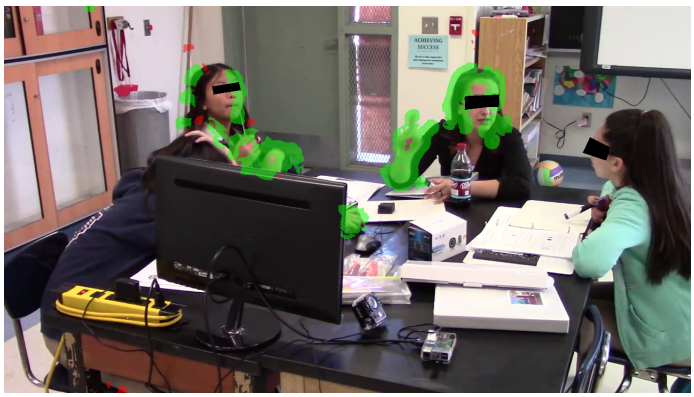
For the dataset to train and validate the component classifier, exemplars were drawn in frames from 8 video clips. Per clip, 25 to 137 frames were used. Component samples from all frames and all clips were scrambled together as the dataset. A random subset of *non-hand* was chosen to train with since there more *non-hand* than *hand* samples.

The random forest classifier out-of-bag accuracy scores are reported in Table 3.1 for approach_1, approach_2, and approach_3 after a random search on some random forest hyperparameters was done.

Classification of components in a frame that came from a video clip whose frames were not used in training is shown in Fig. 3.25. While the moving hands are captured, there are also components over the faces and over the cabinet that are classified as *hand* though they should be classified as *non-hand*.

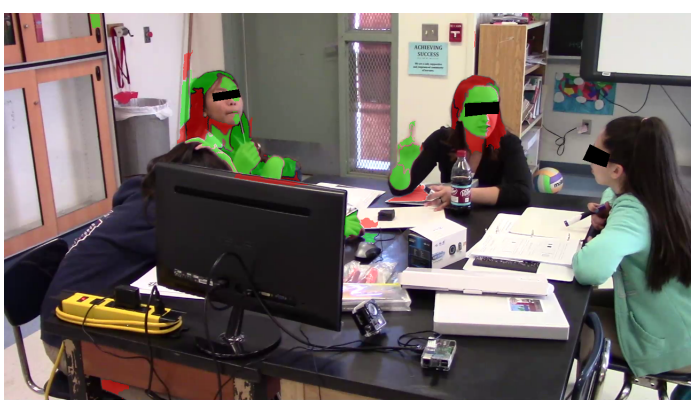
	approach_1	approach_2	approach_3
OOB accuracy	0.7795	0.7468	0.7672

Table 3.1: Out-of-bag accuracy



(a) approach_1

(b) approach_2



(c) approach_3

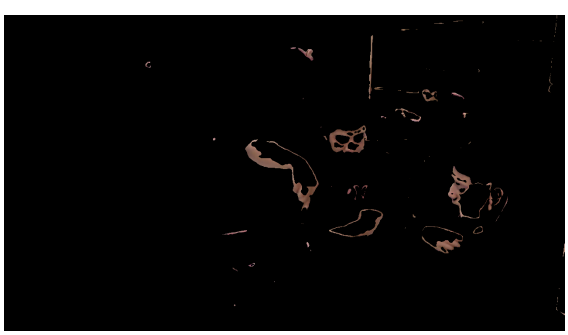
Figure 3.25: Classification of components in an unseen frame. Green means the random forest classifier predicts the component as *hand*. Red means the random forest classifier predicted the component as *non-hand*.

3.4 Patch Skin Region Detection

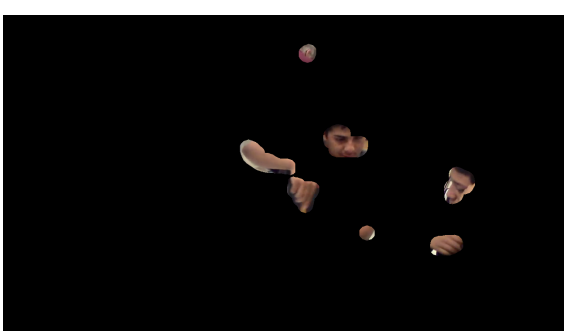
The motivation for making a new skin region detector came from the need to improve prior methods. See how in Fig. 3.26b and Fig. 3.26c the methods used tend to capture the edges of hands and forearms but not the middle regions.

(a) frame

(b) [12][2]



(c) [9] before cleanup



(d) [9] after cleanup

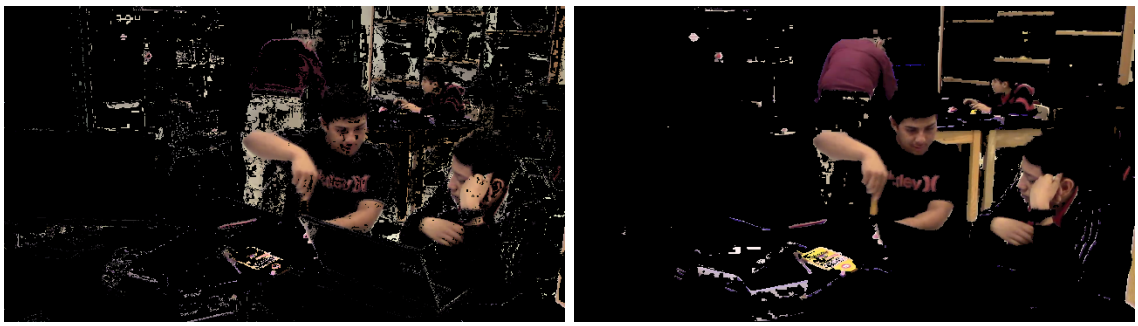
Figure 3.26: The skin regions result on the frame in Fig. 3.26a from the method used by [12][2] is shown in Fig. 3.26b. The skin region result on Fig. 3.26a from the method used by [9] is in Fig.3.26c, the result before cleanup, and in Fig. 3.26d, the result after cleanup.

3.4.1 Method of Patch Skin Region Detector

The method used was to classify 2 x 2 patches of frame as *skin* versus *non_skin*. Thus the feature vector has 12 (2 x 2 x 3) features. Hand region of interest (ROI) are clipped and non-hand regions are clipped as shown in Fig. 3.27 for forming labeled patches for training. Predictions of a k-nearest neighbors (KNN) classifier (k = 5) and a logistic regression classifier were combined via a logical and because the classifiers made different kinds of errors as seen in Fig. 3.28. Cleanup of the predicted skin regions is done according to Fig. 3.29.

(a) hand (b) non_skin

Figure 3.27: Fig. 3.27a is an example of hand regions for forming hand training samples. Fig. 3.27b is an example of non-skin regions for forming non-skin training samples.



(a) KNN (b) logistic regression

Figure 3.28: Fig. 3.28a has skin prediction by KNN classifier. Fig. 3.28b has skin prediction by logistic regression classifier.

function SKIN_CLEANUP(`skin_region_mask`)

`kernel` ← [[0, 0, 0, 1, 0, 0, 0],
[0, 1, 1, 1, 1, 1, 0],
[1, 1, 1, 1, 1, 1, 1],
[1, 1, 1, 1, 1, 1, 1],
[1, 1, 1, 1, 1, 1, 1],
[0, 1, 1, 1, 1, 1, 0],
[0, 0, 0, 1, 0, 0, 0]]
`cleaned` ← OPEN(`skin_region_mask`, `kernel`)
`cleaned` ← **fill holes** in `cleaned`
`cleaned` ← **remove** components in `cleaned` with area ≤ 132 pixels
`kernel` ← [[0, 1, 0],
[1, 1, 1],
[0, 1, 0]]
`cleaned` ← DILATE(`cleaned`, `kernel`)

return `cleaned`
end function

Figure 3.29: Steps taken to clean the skin regions predicted by classifier

3.4.2 Results of Patch Classification for Patch Skin Region Detector

An example of the skin predictions on 2 x 2 patches and of the cleaned result is shown in Fig. 3.30. The training set consisted of 13 frames from different video clips. Leave-one-out (LOO) validation was performed over the training frames. The LOO results are in Table 3.2.

(a) frame (b) predictions

(c) predictions cleaned

Figure 3.30: An example of patch classification and post cleaning.

	mean	std	min	Q1	median	Q3	max	IQR
Precision	0.296	0.111	0.127	0.210	0.273	0.394	0.522	0.184
Sensitivity	0.852	0.137	0.475	0.797	0.893	0.932	0.986	0.135
Specificity	0.956	0.027	0.871	0.952	0.962	0.971	0.983	0.019

Table 3.2: LOO per frame for the labeled regions for the patch skin region classifier.

3.5 Space-Time Exemplars

A premise of this thesis was that optical flow over a duration of time could be used for detection. With this in mind, a reference set was formed on clips of video. Exemplar boxes were drawn over the space region in which a hand passed through during the duration of a time segment. Three second segments were used in the reference. Thus, for the sampled video described in 3.2, there are 45 frames in a 3 second segment. An example of exemplar boxes in a segment is shown in Fig. 3.31.

(a) 180 (b) 208

(c) 276 (d) 292

(e) 340 (f) 356

Figure 3.31: There are three exemplar boxes in this three second segment. Six of the forty-five frames are shown here. Frame numbers are shown in each sub figure.

3.6 Space-Time Component Exploration

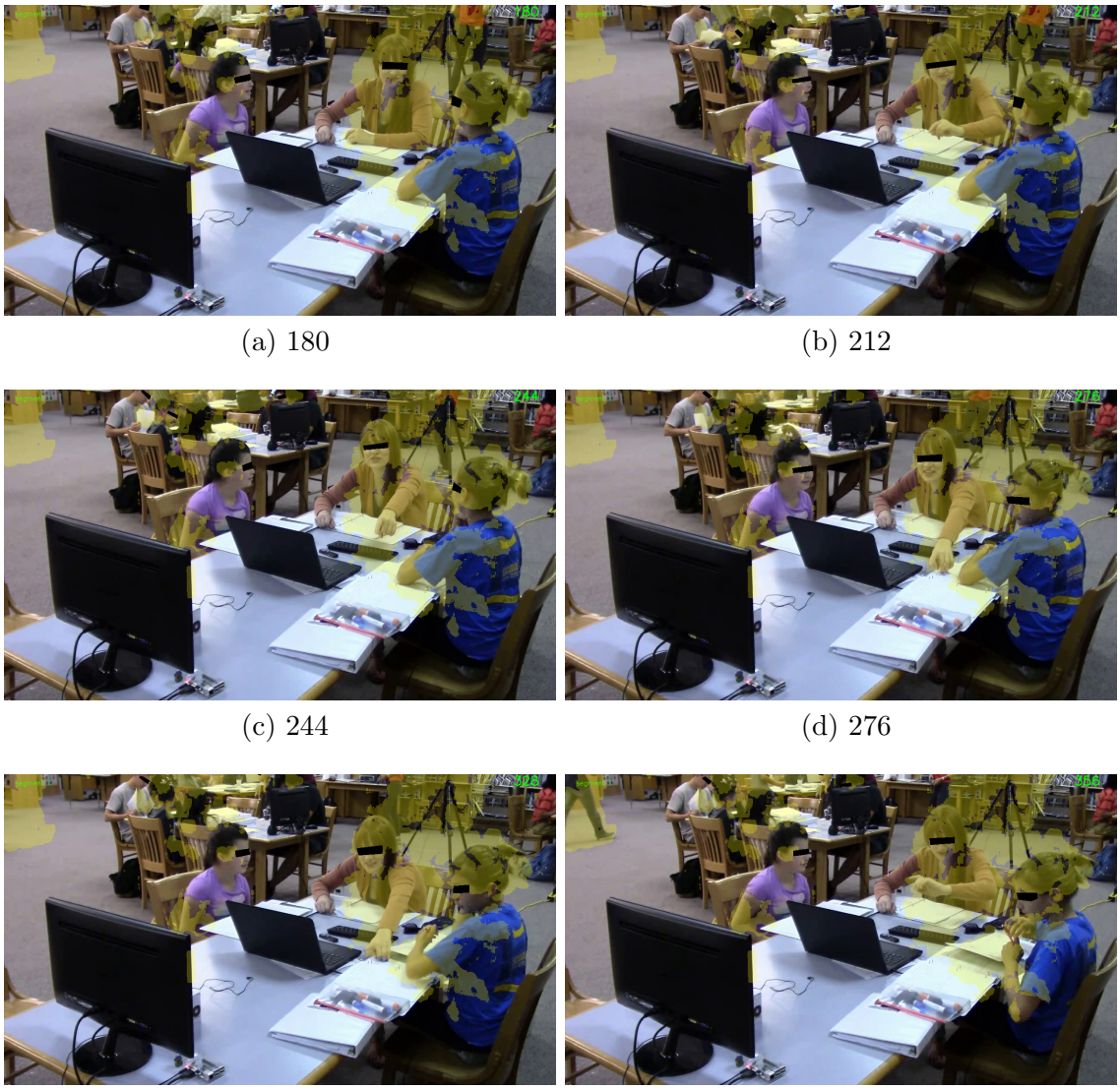
This thesis explores projection through time of components in frames. In theory, the larger projection components formed could be used as regions of interest over which optical flow information over a duration of time could be collected. For this exploration, only one video clip was looked at and the following methods described were formed from observations of that one clip.

Fig. 3.32 describes a first step of how the projection components could formed, and Fig. 3.33 shows an example how it looks in a video segment. Notice in Fig. 3.33, how the projection component covers the facilitator pointing and the student's hand movement; however, the component covers parts of their heads and the person walking in the background as well. Thus, it would be good if the projected component could be broken up.

Fig. 3.34 outlines the steps to break up components and Fig. 3.35 and Fig. 3.36 shows an example. A skin region mask should also help define a region of interest. The projection of skin regions can be formed by taking the union of all the skin region masks for all the frames in a segment. Fig. 3.38a shows an example of the skin projection by itself, and Fig. 3.38b shows an example of both the projection from flow magnitude and from skin region overlapping. Fig. 3.39 gives steps that make a component mask based on how the skin union and broken union overlap. The resulting `component_mask` is shown in Fig. 3.40. The method resulted in components that were more broken up than the original projection component. Fig. 3.41, shows a case where the method did not result in successfully broken components as the face movement and person walking in the background fell under the same component as the hand movement.

```
function PROJECTION(segment_frames)

    for each frame in segments_frames do
        flow_mag_components ← take regions with
                              flow magnitude >
                              MEDIAN(top 3.125% of frame flow magnitude
                                     values)
        flow_mag_components_collect ← append flow_mag_components
    end for

    union ← UNION(all frames in flow_mag_components_collect)

    return union
end function
```

Figure 3.32: Projection by taking union of components formed from applying a threshold to flow magnitude.



(a) 180 (b) 212

(c) 244 (d) 276

(e) 328 (f) 356

Figure 3.33: 6 of the 45 frames in the segment are shown here. The projection found according to Fig. 3.32 is overlaid in yellow. Frame numbers are shown in each sub figure.

function BREAKING(`segment_frames`, `union`)

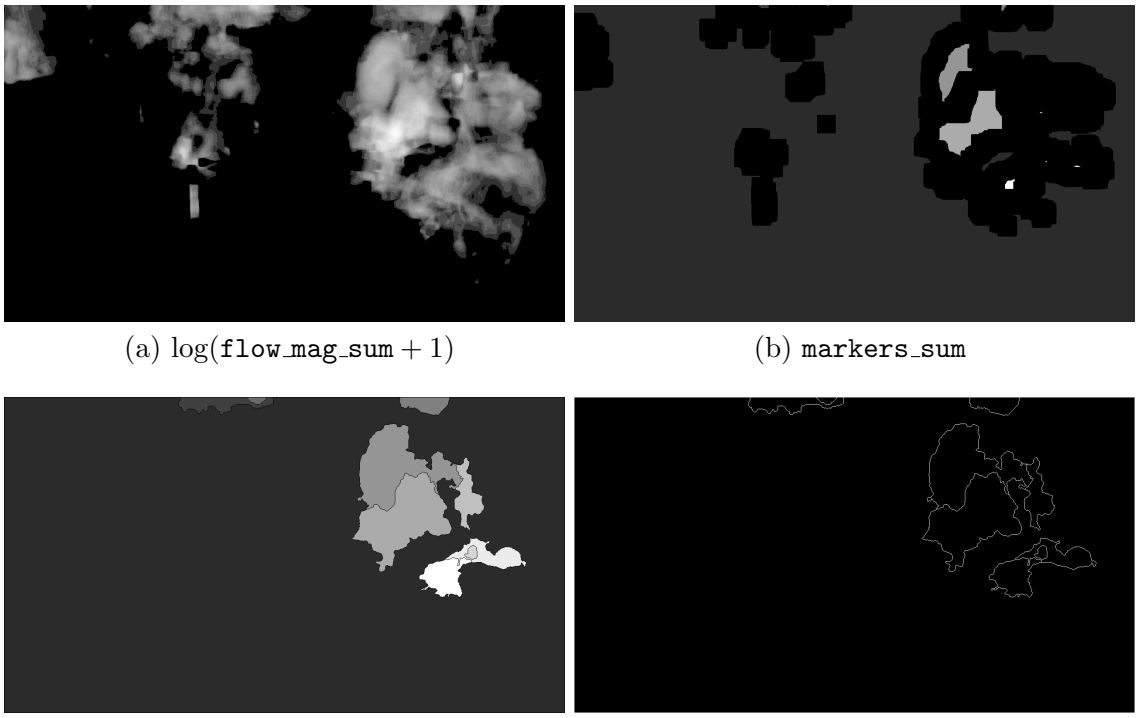
`flow_mag_sum` ← **sum** all the frames of flow magnitude into one frame
`markers_sum` ← **get** unknown region in the markers by eroding and dilating `flow_mag_sum`
`markers_sum_result` ← WATERSHED(flow_mag_sum, markers_sum)
`lines` ← **extract** lines from `markers_sum_result`

`dt_union` ← **do** distance transform on `union`
`markers` ← **get** the unknown region in `markers` as dilation around `dt_union` with a threshold of > 10.0 applied
`markers_result` ← WATERSHED(`dt_union`, `markers`)
`watershed_components` ← **extract** from `markers_result`

`broken` ← **break** `watershed_components` with `lines`

return `broken`
end function

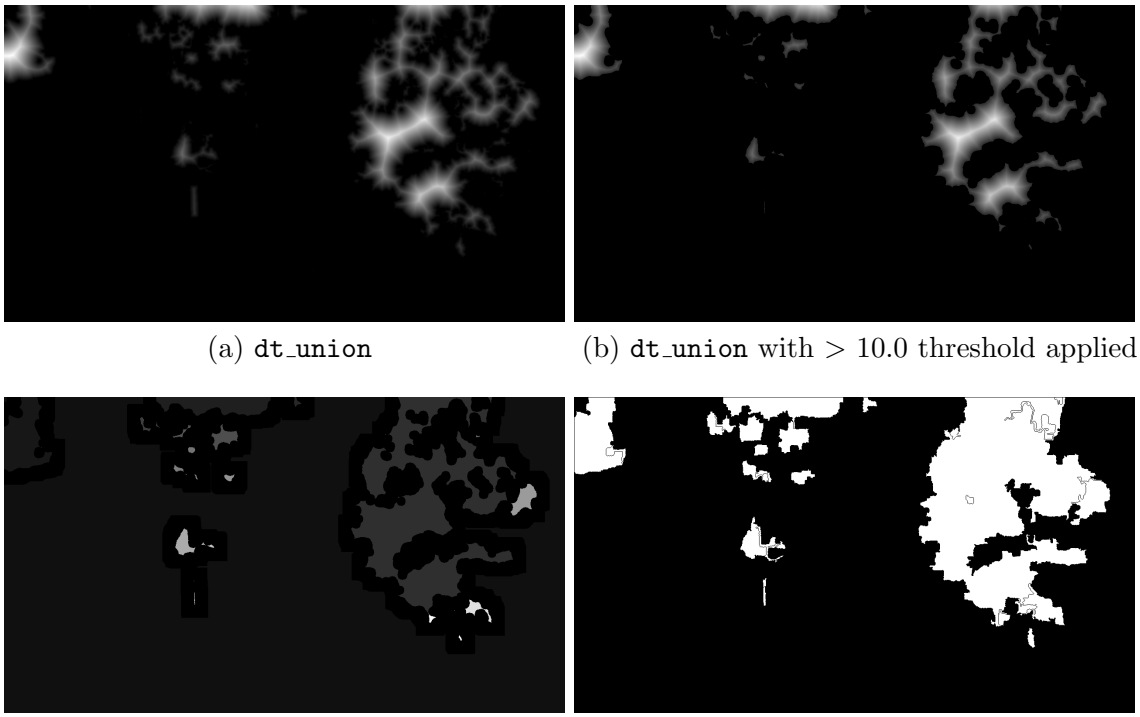
Figure 3.34: Steps to break up projection components.



(a) $\log(\texttt{flow\_mag\_sum} + 1)$ (b) `markers_sum`

(c) `markers_sum_results` (d) `lines`

Figure 3.35: This shows the first group of steps in Fig. 3.34 shown on the union components of the Fig. 3.33 example.



(a) `dt_union` (b) `dt_union` with > 10.0 threshold applied

(c) `markers` (d) `watershed_components`

Figure 3.36: This shows the second group of steps in Fig. 3.34 shown on the union components of the Fig. 3.33 example.

Figure 3.37: This shows the last step, `broken` components (yellow overlay), of Fig. 3.34 shown on the union components of the Fig. 3.33 example.

(a) skin union is red overlay

(b) skin union and `broken` overlap is orange

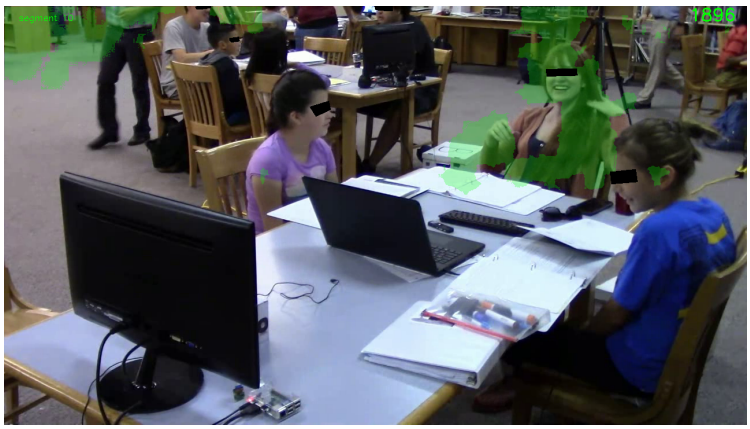
Figure 3.38: Fig. 3.38a shows skin region mask union overlaid as red for the example common to Fig. 3.33 and Fig. 3.37. Fig. 3.38b has the skin union (red) and the broken union (yellow) overlapping (orange).

```
function GET_COMPONENTS(broken, skin_union)

    for each component in broken do
        if component overlaps skin_union by < 50% of component then
            components_mask ← add overlap to components_mask
        else if component overlaps skin_union by ≥ 50% of component then
            components_mask ← add component to components_mask
        end if
    end for

    return components_mask
end function
```

Figure 3.39: Steps to use both `skin_union` and `broken` to form `components_mask`.

Figure 3.40: The figure shows the resulting component mask for segment 1 after the GET_COMPONENTS function of Fig. 3.39 is used.

(a) segment 10 frame 1800



(b) segment 10 frame 1896

(c) segment 10 frame 1976

Figure 3.41: The figure shows the resulting component mask for segment 10 after the GET_COMPONENTS function of Fig. 3.39 is used.

3.7 Space-Time Patches Approach

The region of interest approach with projection components explored in Section 3.6 was not further considered. Instead, a holistic approach which used space-time patches was used for collecting features over time for classification.

The space-time patches were 76 x 76 with 50% overlap in space and covered 3 seconds (45 frames in the sampled video). The space-time patches were given assigned labels depending on how they overlapped the space-time exemplars. The procedure is given in Fig. 3.42 and an example of the labeled patches is in Fig. 3.43. The system for predicting the label (*non-hand* or *hand*) for the space-time patches is depicted in Fig. 3.44.

The skin information used per patch is the number of nonzero pixels (referred to as `skin_count`) in the through time skin overlay image. For every frame in the segment, a skin region image is computed, (for example see Fig. 3.45c and Fig. 3.45d). These are processed into the skin overlay; an example visualizing overlaying is in Fig. 3.45. The result for a segment is in Fig. 3.45f. The basic idea is to project the skin region detection results.

The optical flow was precomputed and saved in videos. To do this the flow magnitude is clipped to be in the range of $[0, 50]$. Then both the flow angle and clipped flow magnitude are rescaled and truncated to be 8 bit unsigned integers. Later, two features are collected per space-time patch from the optical flow as follows:

1. Sum of all 76 x 76 x 45 optical flow magnitude values (referred to as `flow_mag_sum_all`).

2. Histogram of flow with 32 bins for all the 76 x 76 x 45 pixels in the space-time patch. Each bin is divided by the number of flow angles that fell in a bin (unless that number is zero). The bin edges start at $-\pi/32$, go counterclockwise

around unit circle, and have a bin size of $\pi/16$.

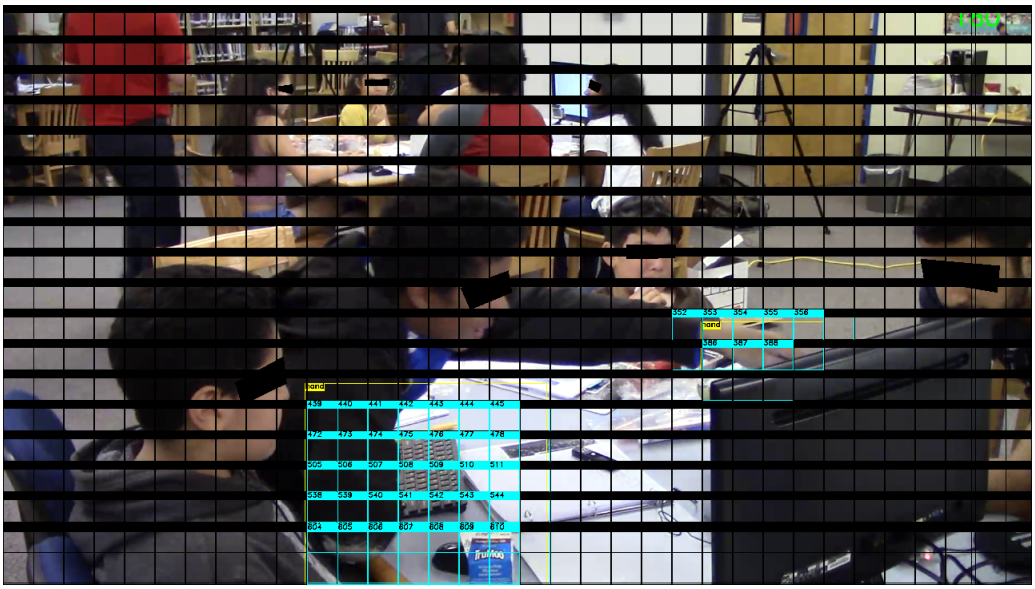
In the Prune Patches block of Fig. 3.44, some patches are predicted as non-hand based on:

$$label = \begin{cases} \text{non-hand} & \textbf{if } \texttt{flow\_mag\_sum\_all} \leq 100 \text{ or} \\ & \quad \texttt{skin\_count}/(76 * 76) \leq 0.1 \\ \text{determine by Trained Classifier} & \textbf{otherwise} \end{cases} \tag{3.1}$$

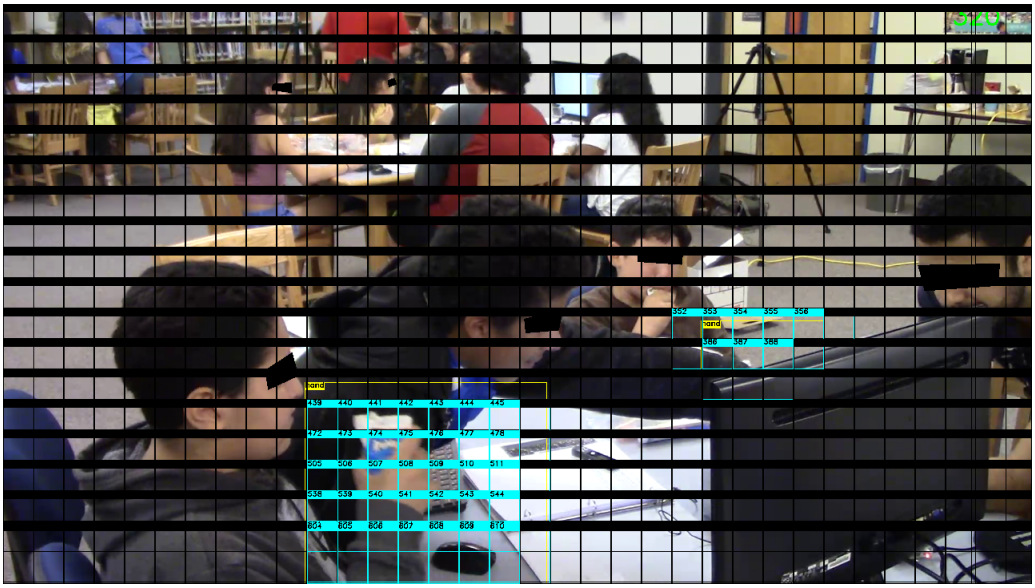
The remaining patches are predicted by a trained classifier. A random forest classifier was used.

```
function LABEL_PATCH(patch, exemplars)
    ▷ initialize
    winning_label ← non-exemplar
    winning_overlap_area / exemplar_area ← 0

    for exemplar in segment's exemplars do
       area_overlap ← get patch and exemplar overlap
       overlap_area / exemplar_area ← get ratio of
                                      area_overlap / exemplar area
       overlap_area / patch_area ← get ratio of
                                   area_overlap / patch area

       if overlap_area / exemplar_area >= 0.2 OR
          overlap_area / patch_area > 0.8 then

          if overlap_area / exemplar_area >
             winning_overlap_area / exemplar_area then

             winning_label← gets the label associated with the exemplar
          end if
       end if
    end for

    return winning_label
end function
```

Figure 3.42: Steps to assign a patch a label from the exemplars in the segment.



(a) 180



(b) 320

Figure 3.43: 2 frames of the 45 frames in a segment. The exemplar boxes are in *yellow*. The patches that take the *non-hand* label are in *black*. The patches that take the *hand* label are in *cyan*.

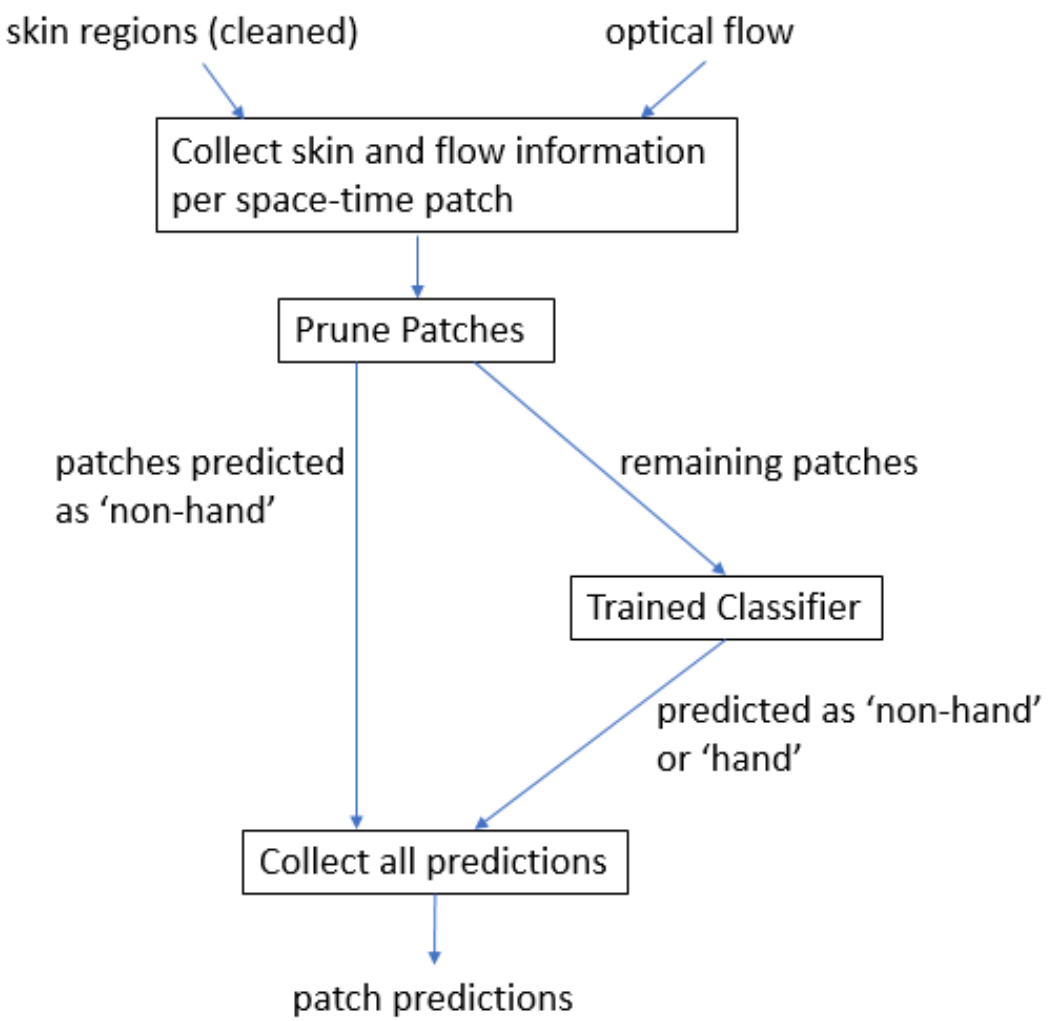
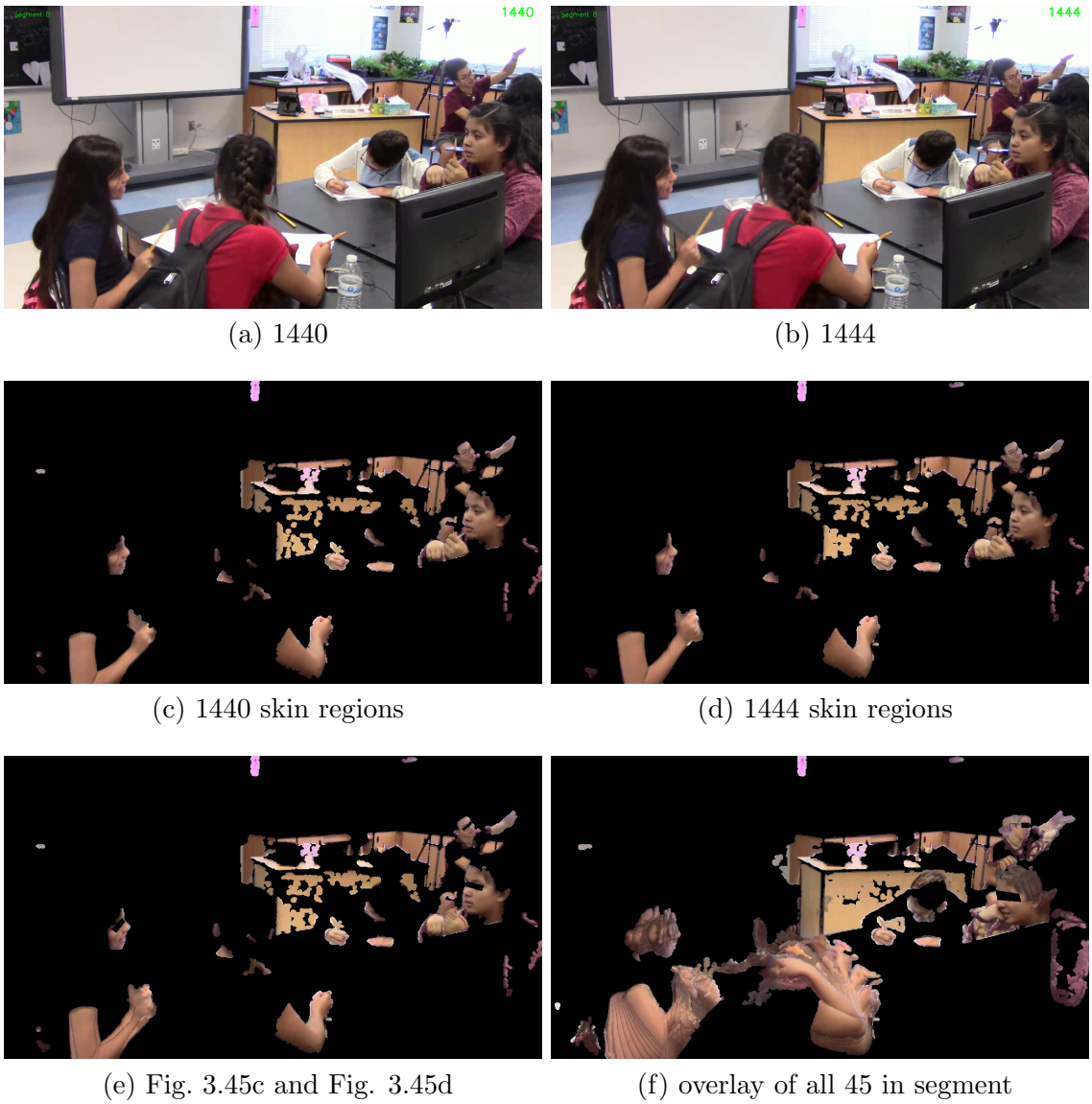


Figure 3.44: This flowchart is for the system used to predict the label on space-time patches.



(a) 1440 (b) 1444

(c) 1440 skin regions (d) 1444 skin regions

(e) Fig. 3.45c and Fig. 3.45d (f) overlay of all 45 in segment

Figure 3.45: This figure shows a visual example of forming skin overlay image. Fig. 3.45a through Fig. 3.45b show two sequential frames of a segment. Fig. 3.45c through Fig. 3.45d show the skin region detections by the Section 3.4 method. Fig. 3.45e shows the result of overlaying Fig. 3.45c and Fig. 3.45d. Lastly, Fig. 3.45f shows the result of overlaying each next frame in a video segment into the previous overlay image. Overlaying here means that the union of frames is taken; where there is intersection, the RGB values are combined with a weight of 0.5.

3.8 Computation

Computation of cleaned skin regions and the optical flow and extraction of the features for the dataset used to validate Section 3.7 was done via an account on the machine Wheeler at the Center for Advanced Research Computing. Wheeler is a SGI ALtixXE, Xeon X5550, Intel Xeon Nehalem EP, 2.67GHz machine with 294 nodes, 8 cores per node, 48 GB RAM / core, and 40TB of local scratch. Memory intensive tests were preformed on a computer belonging to the Image and Video Processing and Communications Laboratory. It was a Dell Precision Tower 7910 with Intel Xeon Processor E5-2630 v4, with 32 GB memory, and with a Windows operating system.

Chapter 4

Results

4.1 Result of Space-Time Patch Classification

The dataset used for validation of the space-time classification consisted of 15 video clips from the AOLME dataset. Ten of the videos were approximately 39 seconds (thus 13 segments per clip), and five of the videos were approximately 99 seconds (33 segments per clip). All of the videos were approximately 60 frames per second. Exemplars were drawn for moving hands in the video clips according to Section 3.5. A frame from each video clip is shown in Fig. 1.1.

Descriptions of the video clips are given in Table 4.1, Table 4.2, Table 4.3, and Table 4.4. Scores for space-time patch classification for leave-one-out (LOO) validation over the video clips are in Table 4.5. The statistics on the LOO validation are in Table 4.6. Fig. 4.1 shows patch classifications for a single video segment.

Video	Content
V1	Student and facilitator are at the primary table. Both face the camera. There is one other group in the far background.
V2	Four students are at the primary table. There is another group in the right back side.
V3	Three students and a facilitator are at the primary table. There are two other groups in the background. People walk around in the background as well.
V4	Two facilitators and two students are at the primary table. There are laptop monitors blocking some movement. One student is standing and wanders to and from the table.
V5	Four student and two facilitators are at the primary table. One of facilitators is standing. People block and the view of others. The monitor blocks activity as well. There are two other groups in the background.
V6	Two students and a facilitator are at the primary table. The facilitator is standing. There is another group and a lone person at a laptop in the background. A person walked in the background.
V7	A facilitator and four students are at the primary table. One student returns to the table during the clip.
V8	Three students and two facilitators are at the primary table. There are two groups in the background. People walk in the background.
V9	Four students are at a table. There are four other groups in the background. People walk in the background.
V10	Two students and a facilitator at the primary table. There are three other groups in the background. People walk in the background.

Table 4.1: This table holds notes about the content in the 39 second clips (V1 - V10).

Video	Content
V11	Two students and a facilitator are at the primary table. There is one other group in the background. There is a person fiddling with a camera in the background. And some other people working on a laptop.
V12	Two students and a facilitator are at the primary table. There are two other groups in the background. People walk in the background at times.
V13	Two students and two facilitators are at the primary table. One of the facilitators moves from a standing by the table to a leaning by the table on a box. There are two groups in the background. There are walking people and people checking equipment in the background.
V14	Four students and a facilitator are at the primary table. People walk through the in the foreground in occasionally. A couple people have an animated conversation while standing in the background.
V15	Two students and two facilitators at the primary table. The facilitators come and leave from the primary table. One of the students leaves partway. There are five other groups in the background. People walk in the background.

Table 4.2: This table holds notes about the content in the 99 second clips (V11 - V15).

Video	Comments on Skin Detection from Cleaned Patch Prediction
V1	Skin, especially the hands, was detected well. There was some box, wood, orange bag, and orange trim that was detected as skin well, but those types of regions were reduced from what was present in the scene.
V2	The skin was detected well. There is sizable amount of cabinet detected as skin.
V3	The skin at the primary table was detected well, though the skin detection on the facilitator's face and upper arm was spotty. There were splotches of wood-like regions captured. There was a light purple shirt that was captured as skin. There were some duller red shirts that were not captured.
V4	The skin detection was missing some regions at times. The faces of two people was not fully detected. There is only a small amount of wood (there are bookshelves with wood trim in the background) being classified as skin.
V5	The skin detection on the hands is okay, but there are some missing splotches. The faces are pretty splotch. There are some non-skin items classified as skin.
V6	The skin detection is pretty good. There is some detection of wood chairs and trim as skin.
V7	The skin of the hands is detected somewhat, but it is broken up sometimes. Skin of face is only partially detected. There is a small amount of wood trim and reddish brown books on the bookshelves that is detected as skin.
V8	Some of the skin detection is good, but some of the skin detection is broken up. There is only a little detection of wood trim as skin.
V9	Skin is detected, but it is somewhat broken up. Faces are especially broken up. Only a small amount of wood trim was detected as skin.
V10	The skin detection is poor in this clip. The skin at the primary table is especially not captured. Bits of a rather light purple shirt are detected as skin. A pinkish rust cardigan is detected as skin. There is some wood trim and reddish part of object that are classified as skin.

Table 4.3: This table holds notes about the patch skin detector in the 39 second clips (V1 - V10).

Video	Comments on Skin Detection from Cleaned Patch Prediction
V11	The skin detection at the primary table does well. There is a large region of wood cabinet that gets detected as skin.
V12	The skin detection for the hands is okay. The skin detection on the face is splotchy at times. There is some wood chairs, brown plastic bag, and reddish part of the background that get detected as skin.
V13	Some of the skin detection is okay, but some is poorly detected. There is some wood and orange sign that get detected as skin.
V14	The skin detection works pretty well, but there is one face that it is very splotchy. There is wood cabinet and bookshelf that are prominently detected as skin.
V15	The skin of the hands is detected pretty well. The detected skin of faces is holey at times. There is a lot of little bits of wood or orange parts of the background that get detected as skin.

Table 4.4: This table holds notes about the patch skin detector in the 99 second clips (V11 - V15).

		For the operating point defined by minimum Euclidean to (Specificity, Sensitivity) = (1.0, 1.0)			
Video	ROC AUC	Euclidean Distance	Specificity	Sensitivity	Accuracy
V1	0.960	0.088	0.923	0.953	0.923
V2	0.889	0.241	0.789	0.880	0.794
V3	0.840	0.316	0.784	0.767	0.783
V4	0.889	0.180	0.916	0.833	0.913
V5	0.930	0.169	0.874	0.886	0.875
V6	0.938	0.174	0.849	0.911	0.853
V7	0.892	0.184	0.836	0.913	0.840
V8	0.903	0.217	0.829	0.857	0.831
V9	0.895	0.237	0.794	0.882	0.802
V10	0.758	0.406	0.823	0.627	0.816
V11	0.910	0.214	0.865	0.834	0.863
V12	0.917	0.188	0.878	0.855	0.877
V13	0.913	0.153	0.880	0.904	0.881
V14	0.879	0.236	0.788	0.895	0.793
V15	0.875	0.246	0.783	0.881	0.785

Table 4.5: Validation results on the 15 AOLME clips.

	μ	σ	Min	Q1	Median	Q3	Max	IQR
ROC AUC	0.892	0.045	0.758	0.879	0.895	0.917	0.960	0.038
Accuracy	0.842	0.045	0.783	0.794	0.840	0.877	0.923	0.083

Table 4.6: Statistics of validation results on the 15 AOLME clips.

Figure 4.1: This figure shows the classification for the first segment of V2. Regions where there are no patches indicate patches got pruned there. Patches that are *gray* were classified non-hand. Patches that are *purple* were classified as hand. The exemplar boxes are in *yellow*.

Chapter 5

Conclusions and Future Work

The final quantitative results of an average accuracy of 84% and ROC AUC 0.89% for space-time patch classification on the LOO of video can be improved. Visual inspection of the results show error on the side of over segmenting, i.e. there is a sizable amount of false positives, non-hand patches classified as hands. This shows that the trained classifier is not able to distinguish between hand movements and other movements. I suspect that the trained classifier may be misclassifying high movements as hand. See Fig. 5.1 and then Fig.5.2. The places where the patches show up correspond to the regions with the most motion over time. This comparison was not investigated extensively, however.

One thought for future work is to use the face detection work developed for the AOLME dataset in [12][13]. This would help reduce the over segmentation that occurs in patches over faces.

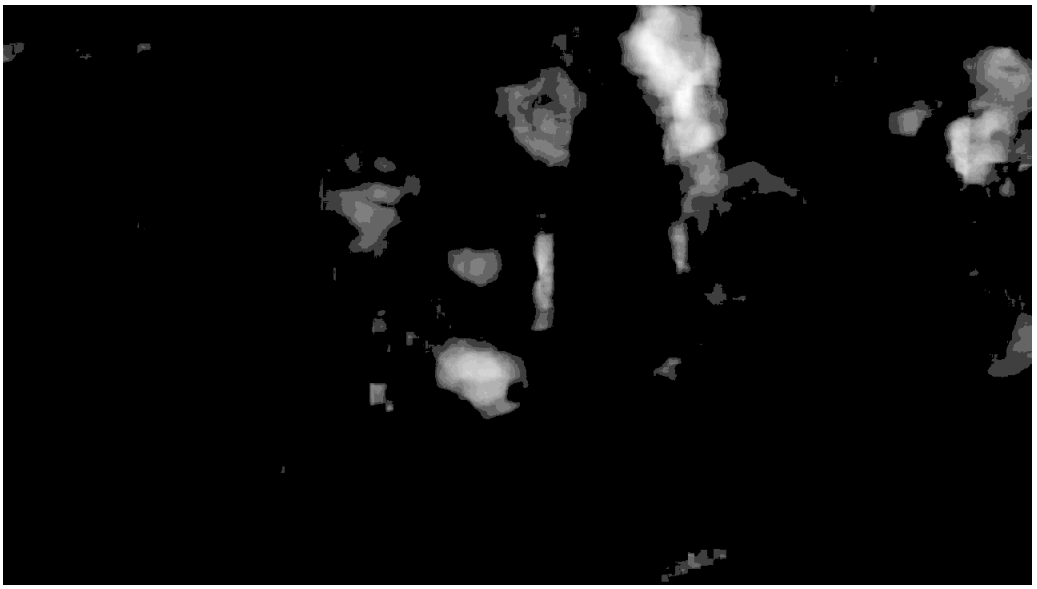


Figure 5.1: Sum of the flow magnitude sum for all the frames under a log transformation.

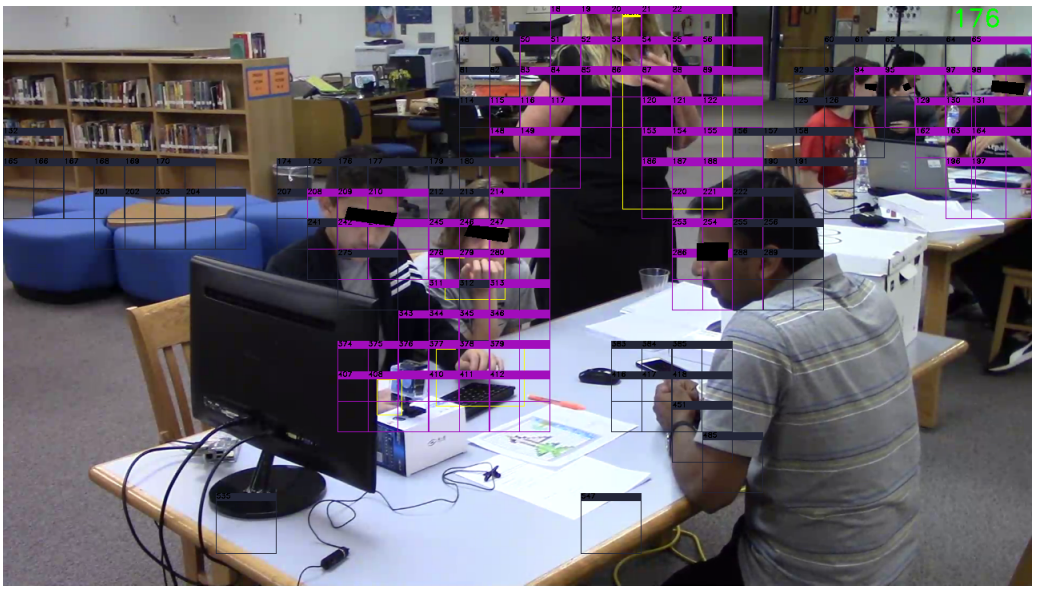


Figure 5.2: This figure shows the classification for the first segment of V13. Regions where there are no patches indicate patches got pruned there. Patches that are *gray* were classified non-hand. Patches that are *purple* were classified as hand. The exemplar boxes are in *yellow*.

References

[1] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 1949–1957. IEEE, 2015.

[2] Nusirwan Anwar bin Abdul Rahman, Kit Chong Wei, and John See. Rgb-h-cbcr skin colour model for human face detection. *Faculty of Information Technology, Multimedia University*, 4, 2007.

[3] Patrick Buehler, Mark Everingham, Daniel P Huttenlocher, and Andrew Zisserman. Long term arm and hand tracking for continuous sign language tv broadcasts. In *Proceedings of the 19th British Machine Vision Conference*, pages 1105–1114. BMVA Press, 2008.

[4] Cody W Eilar, Venkatesh Jatla, Marios S Pattichis, Carlos LópezLeiva, and Sylvia Celedón-Pattichis. Distributed video analysis for the advancing out of school learning in mathematics and engineering project. In *Signals, Systems and Computers, 2016 50th Asilomar Conference on*, pages 571–575. IEEE, 2016.

[5] Cody Wilson Eilar. Distributed and scalable video analysis architecture for human activity recognition using cloud services. MSc thesis, University of New Mexico, 2016.

[6] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003.

[7] Rong Feng, Camilo Perez, and Hong Zhang. Towards transferring grasping from human to robot with rgbd hand detection. In *Computer and Robot Vision (CRV), 2017 14th Conference on*, pages 285–291. IEEE, 2017.

[8] Abigail Jacoby, Marios S. Pattichis, Sylvia Celedon-Pattichis, and Carlos LopezLeiva. Context-sensitive human activity classification in collaborative learning environments. In *IEEE Southwest Symposium on Image Analysis and Interpretation*, in press, 2018.

[9] Abigail Ruth Jacoby. Context-sensitive human activity classiffcation in video utilizing object recognition and motion estimation. MSc thesis, University of New Mexico, 2018.

[10] Arpit Mittal, Andrew Zisserman, and Philip HS Torr. Hand detection using multiple proposals. In *BMVC*, pages 1–11. Citeseer, 2011.

[11] Kankana Roy, Aparna Mohanty, and Rajiv R Sahay. Deep learning based hand detection in cluttered environment using skin segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 640–649, 2017.

[12] Wenjing Shi. Human attention detection using am-fm representations. MSc thesis, University of New Mexico, 2016.

[13] Wenjing Shi, Marios S. Pattichis, Sylvia Celedon-Pattichis, and Carlos LopezLeiva. Robust head detection in collaborative learning environments using am-fm representations. In *IEEE Southwest Symposium on Image Analysis and Interpretation*, in press, 2018.

[14] Eungyeol Song, Hyeongmin Lee, Jaesung Choi, and Sangyoun Lee. Ahd: Thermal image-based adaptive hand detection for enhanced tracking system. *IEEE Access*, 6:12156–12166, 2018.