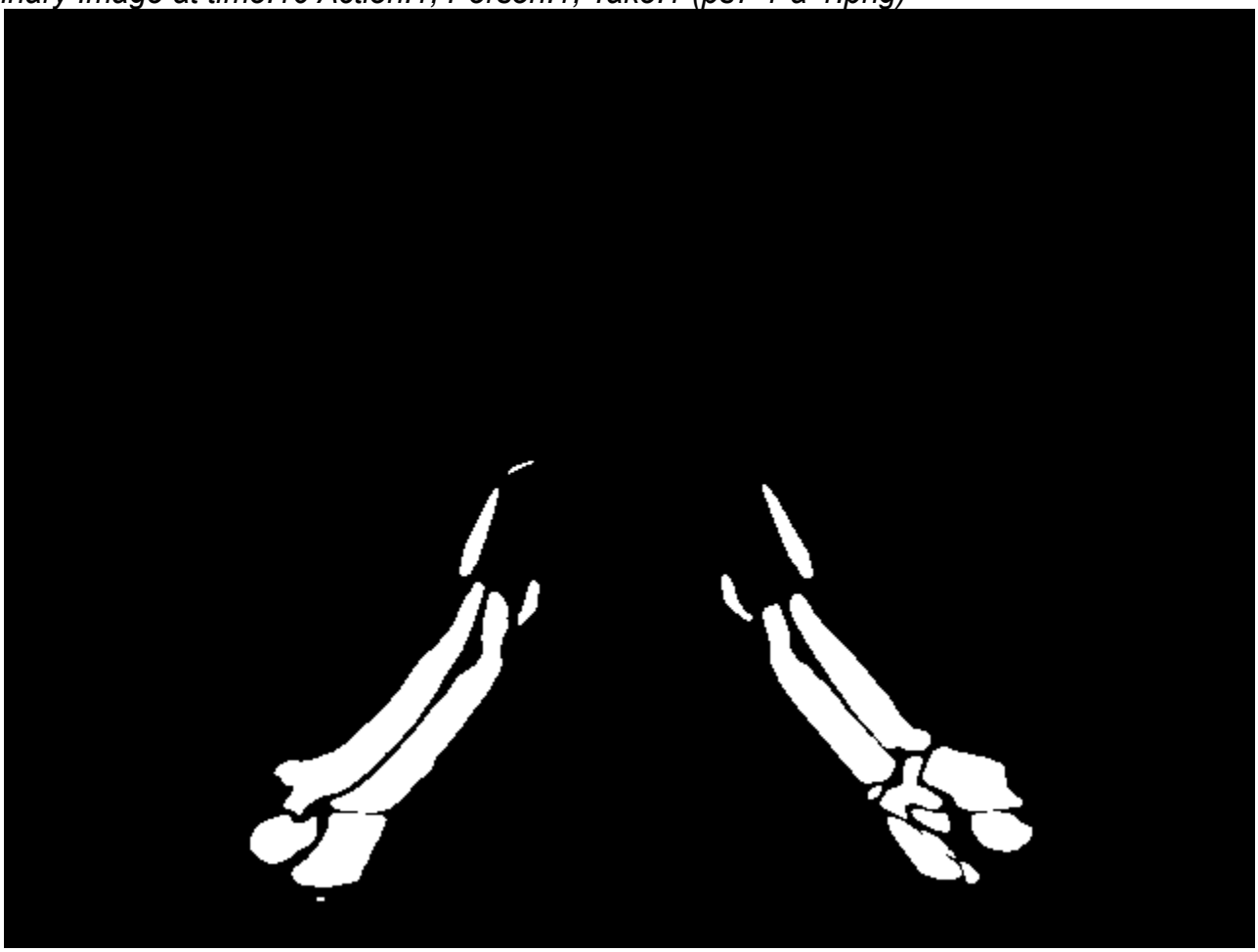
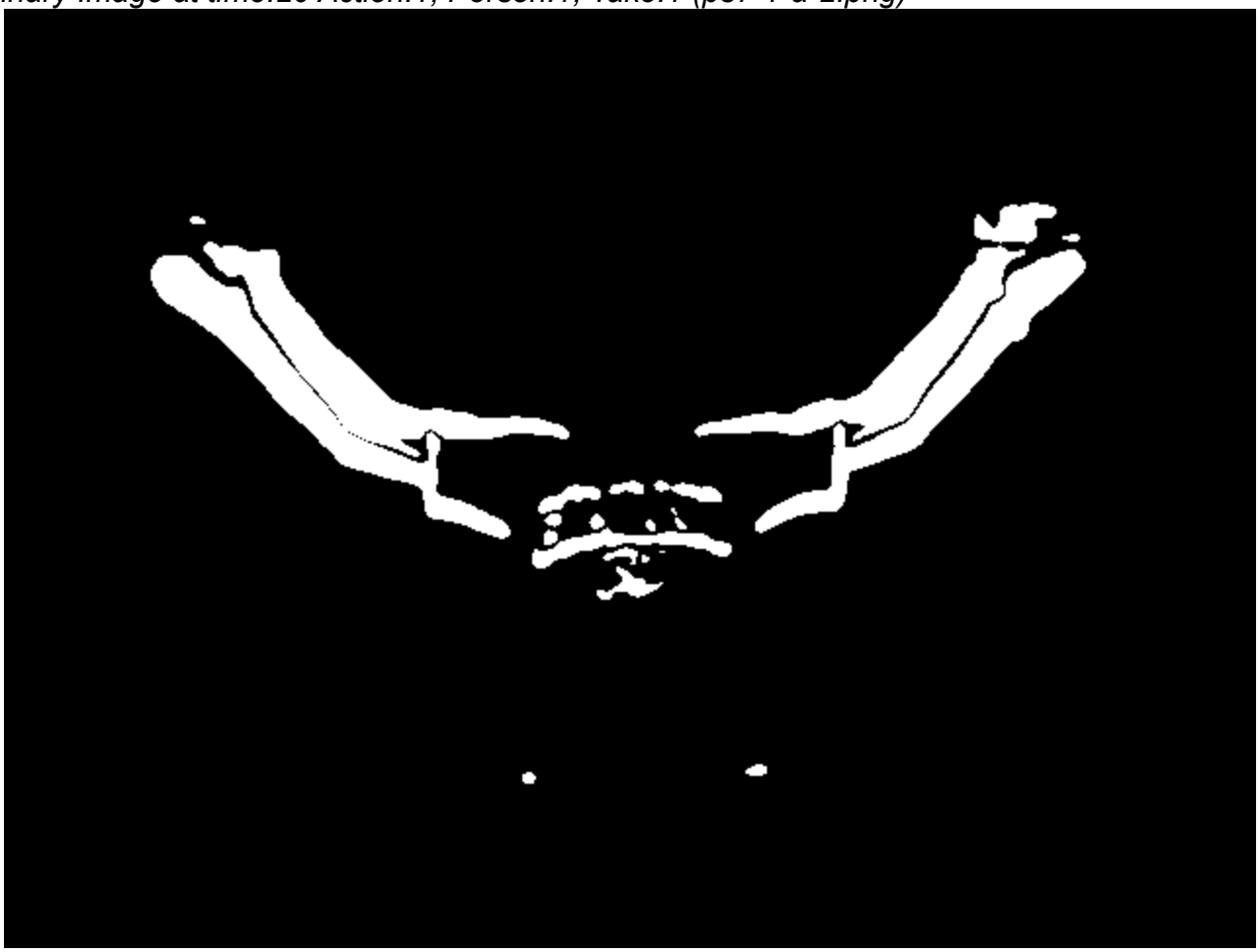
Jonathan Hudgins
Problem Set 7: Motion History Images
CS4495, Spring 2015 OMS
GTID: 903050550

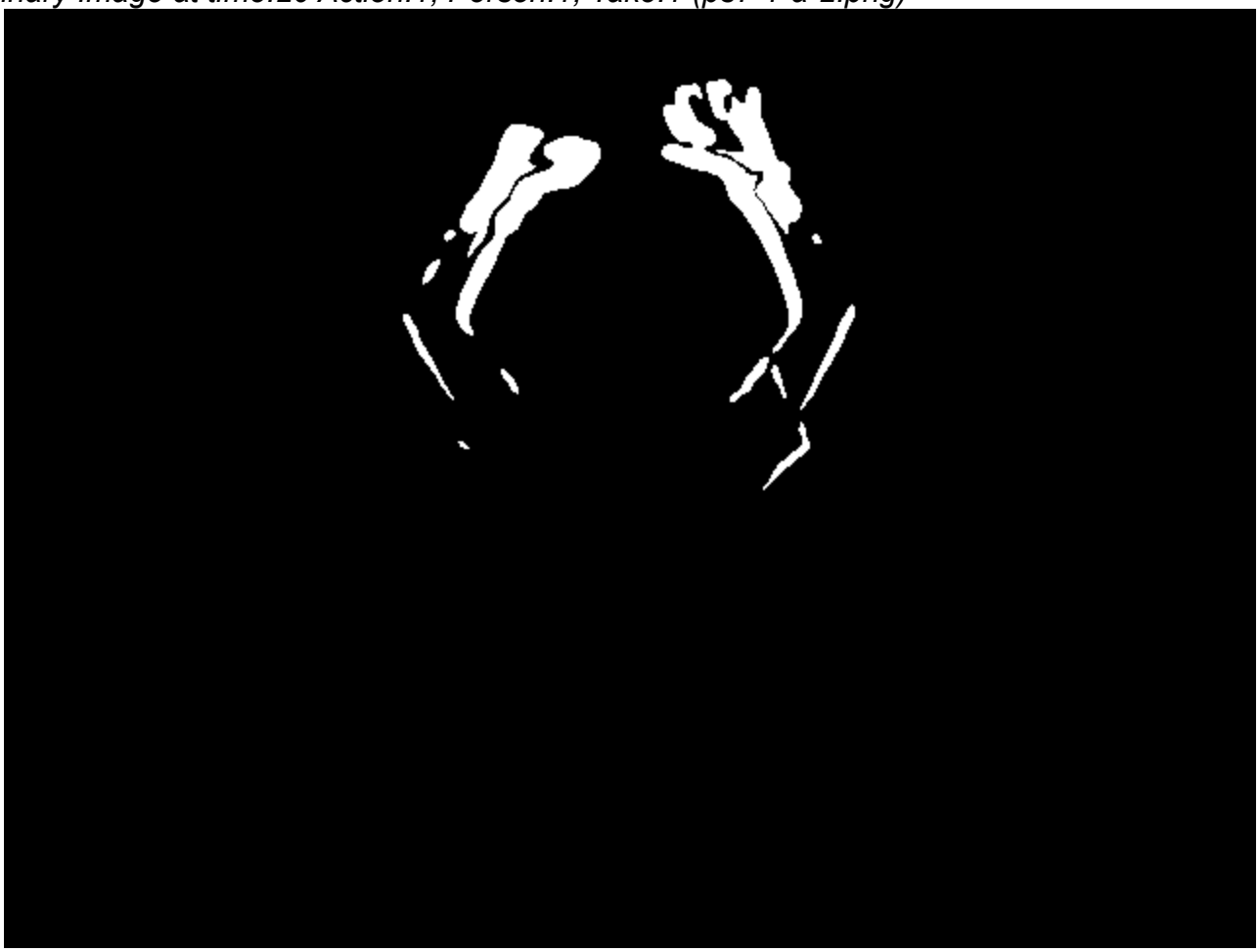
Part 1a:
Binary Image at time:10 Action:1, Person:1, Take:1 (ps7-1-a-1.png)



Binary Image at time:20 Action:1, Person:1, Take:1 (ps7-1-a-2.png)



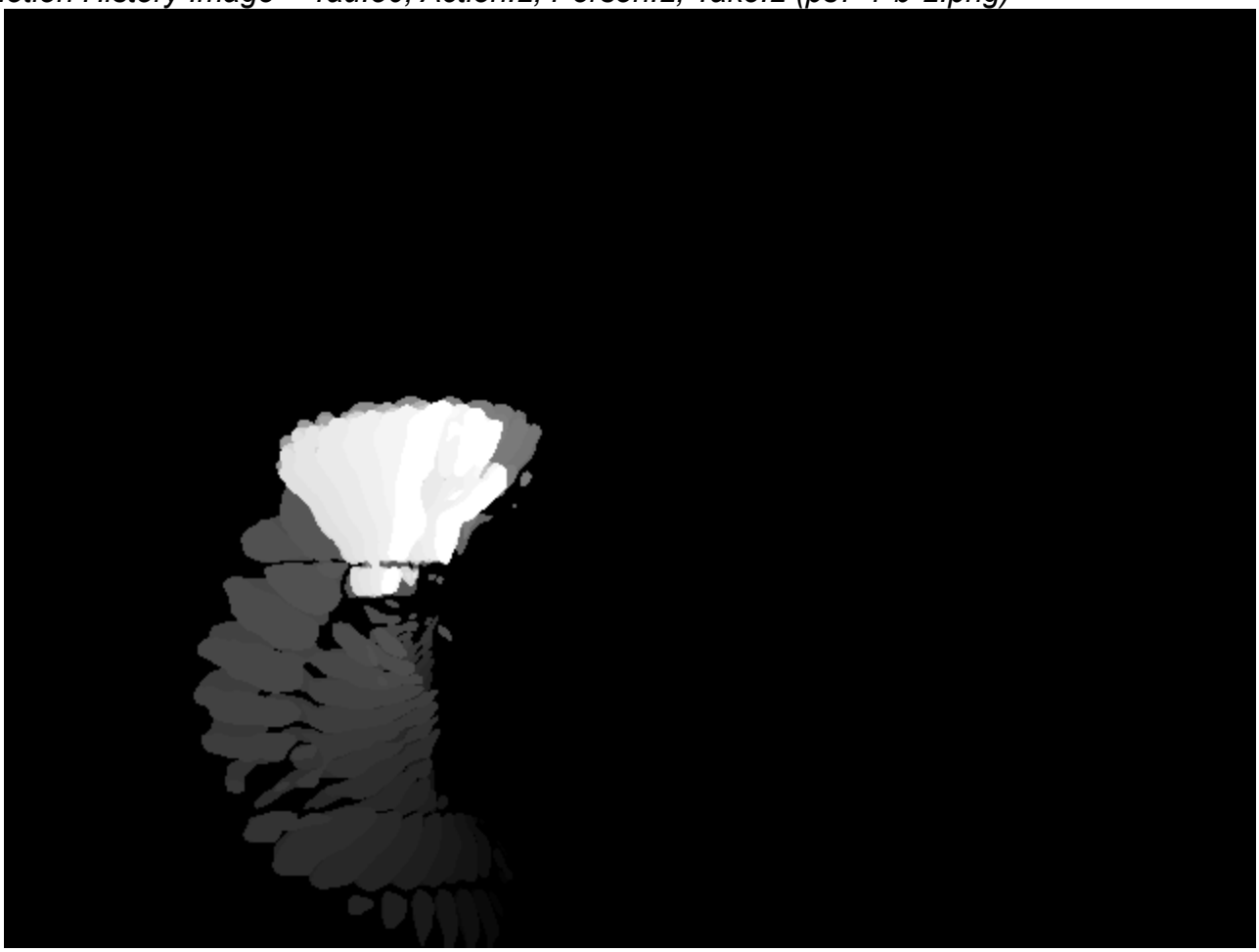
Binary Image at time:20 Action:1, Person:1, Take:1 (ps7-1-a-2.png)



Part 1b:

Motion History Image – Tau:30, Action:1, Person:2, Take:2 (ps7-1-b-1.png)

Motion History Image – Tau:50, Action:2, Person:2, Take:2 (ps7-1-b-2.png)



Motion History Image – Tau:30, Action:3, Person:2, Take:2 (ps7-1-b-3.png)

I used the second take of the second person for each action because it seemed to give more representative results.

I used:

- tau = 30 for action 1
- tau = 50 for action 2
- tau = 30 for action 3

Part 2a:

Central Moment Confusion Matrix:
[[9. 0. 0.]
 [0. 9. 1.]
 [0. 0. 8.]]

Scaled Confusion Matrix:
[[9. 0. 0.]
 [0. 9. 1.]
 [0. 0. 8.]]

To weight the dimensions I independently normalized each dimension to fall between 0 and 1.
Normalized Central Moment Confusion Matrix:
[[9. 0. 0.]
 [0. 9. 1.]
 [0. 0. 8.]]

Normalized Scaled Confusion Matrix:
[[9. 0. 0.]
 [0. 9. 1.]
 [0. 0. 8.]]

Surprisingly the non-normalized matrices show near perfect results. There is only one of 27 mis-categorized action. It seems that further tweaking would have diminshing returns so I focused on part 2b.

Part 2b:
Removing all actions/takes for the person we are testing returns the following results.
Scale Invariant Moment Confusion Matrix (after removing person from training set):
Person 1:
[[3. 0. 0.]
 [0. 3. 1.]
 [0. 0. 2.]]

Person 2:
[[3. 0. 0.]
 [0. 3. 0.]
 [0. 0. 3.]]

Person 3:
[[3. 0. 0.]
 [0. 3. 0.]
 [0. 0. 3.]]

Average:
[[3.   0.   0.   ]
 [0.   3.   0.33333333]
 [0.   0.   2.66666667]]

These results are identical to Part 2a. It looks like the *sigma*, *theta* and *tau* choices are very effective!

Normalizing the dimensions provides some interesting results:
Normalized Scale Invariant Moment Confusion Matrix (after removing person from training set):
Person 1:
[[3. 0. 3.]
 [0. 3. 0.]
 [0. 0. 0.]]

Person 2:
[[3. 0. 2.]
 [0. 3. 0.]
 [0. 0. 1.]]

Person 3:
[[0. 0. 0.]
 [0. 3. 0.]
 [3. 0. 3.]]

Person Average:
[[2.   0.   1.66666667]
 [0.   3.   0.   ]
 [1.   0.   1.33333333]]

So the scaling already in the dimesions does a better job than normalized values. With such a small training set, it is difficult to predict if this holds true in general or just for these people and actions.