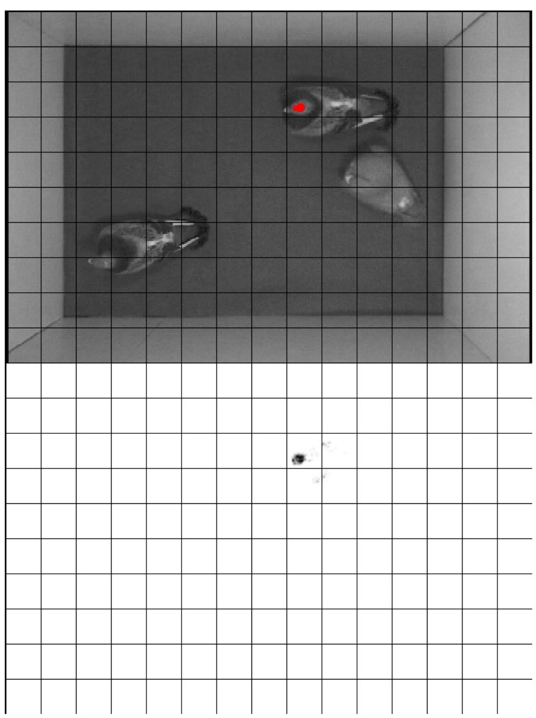
University of Toronto

Particle Filtering

CSC2503 Fall 2014
Foundations of Computer Vision

Alexander Hong
(c4hongal 997584706)



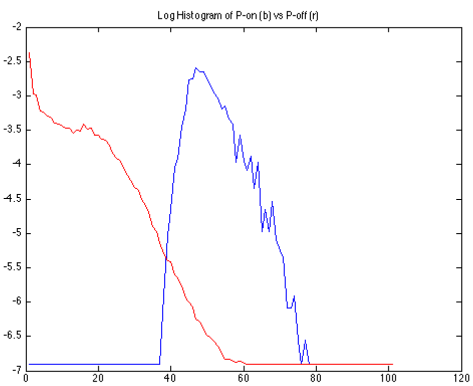
November 21, 2014

Contents

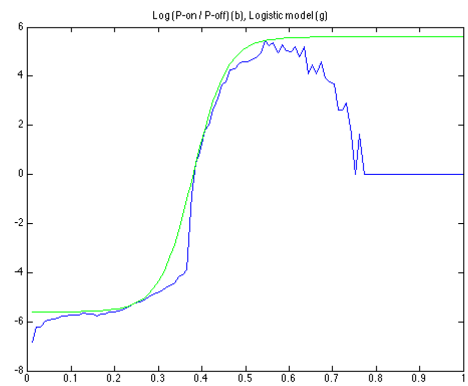
1 Log odds

After grabbing and training for the appropriate HOG response for the targets of interest, the log histogram of p_{On} and p_{Off}, and log likelihood ratio $L(z)$ for each target is generated, as shown in Figure 1. The log likelihood graphs, however, do not fit to a single logistic model over the whole range $z \in [0, 1]$. The most important part of the graphs is where p_{On} and p_{Off} diverge in the log histogram graphs because it is the portion describing the difference between positive and negative training samples. Hence, the most important portion of the log likelihood graph to fit is from $z = 0$ to somewhere after the global peak of the likelihood and before the convergence point because it is where noticable difference between p_{On} and p_{Off} is observed. For instance, the most important portion to fit is $z \in [0, 0.7]$ for Figure 1b, $z \in [0, 0.6]$ for Figure 1d, $z \in [0, 0.275]$ for Figure 1f, and $z \in [0, 0.7]$ for Figure 1h.

The most ideal situation is to have the log histogram of p_{On} and p_{Off} as far away from each other as possible (i.e. greater difference), since further separation would make the tracking easier due to greater distinct differences between positive and negative classes. For example, the red pigeon's back, Figure 1c, would be much easier to track than the red pigeon's head, Figure 1e. Figure 1e shows greater ambiguity due to greater overlap of positive and negative training samples. In other words, the training samples are similar, which reduces the accuracy of the tracking system.

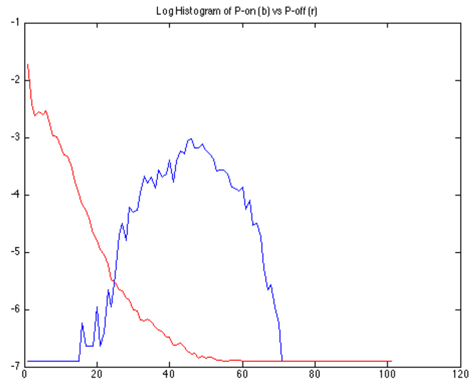


(a) Grey pigeon back log histogram

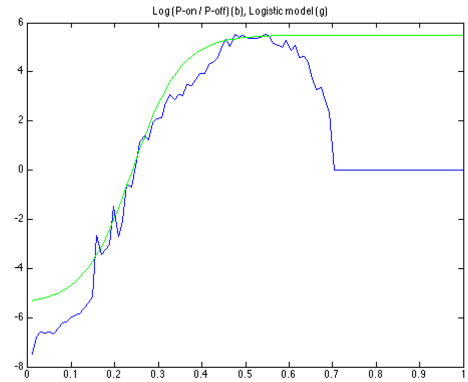


(b) Grey pigeon back log likelihood ratio

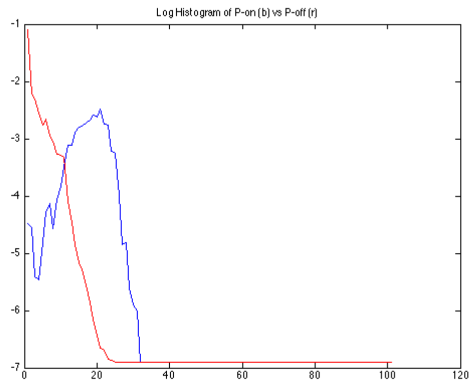
Figure 1: Log Histogram and log likelihood ratio of targets of interest



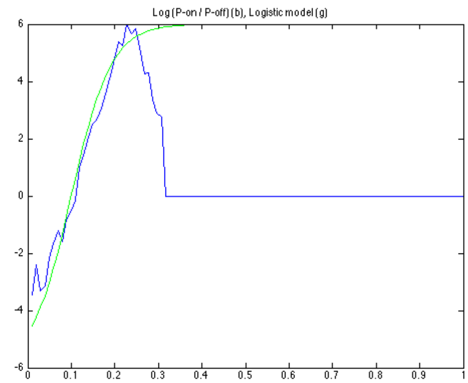
(c) Red pigeon back log histogram



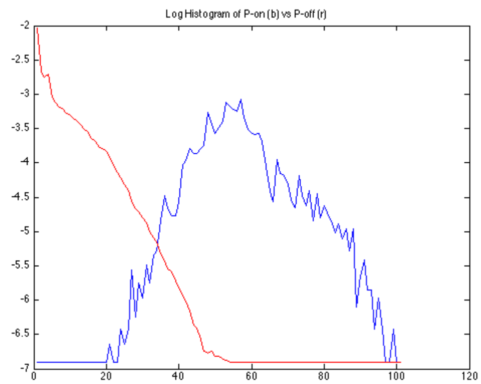
(d) Red pigeon back log likelihood ratio



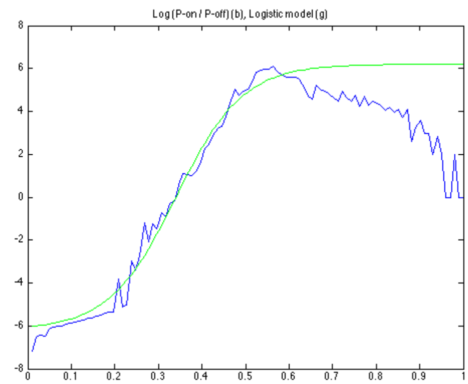
(e) Red pigeon head log histogram



(f) Red pigeon head log likelihood ratio



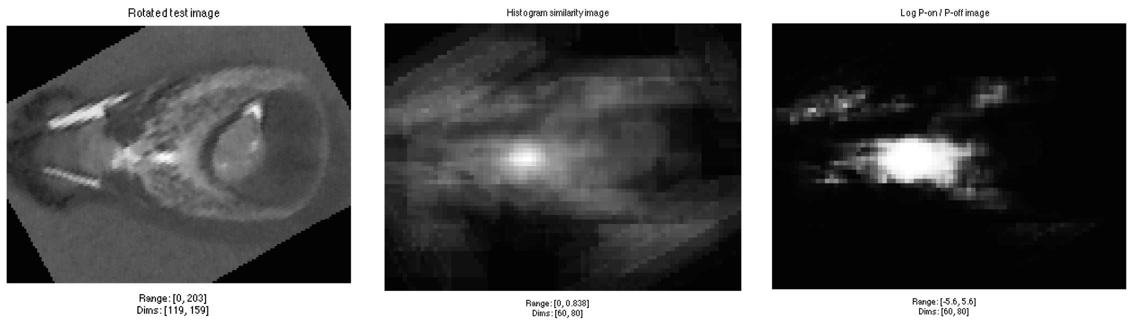
(g) Grey pigeon head log histogram



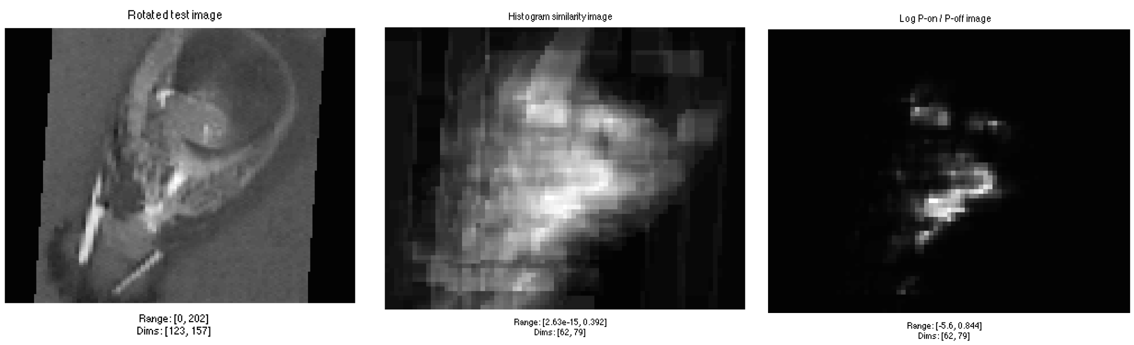
(h) Grey pigeon head log likelihood ratio

Figure 1: Log Histogram and log likelihood ratio of targets of interest

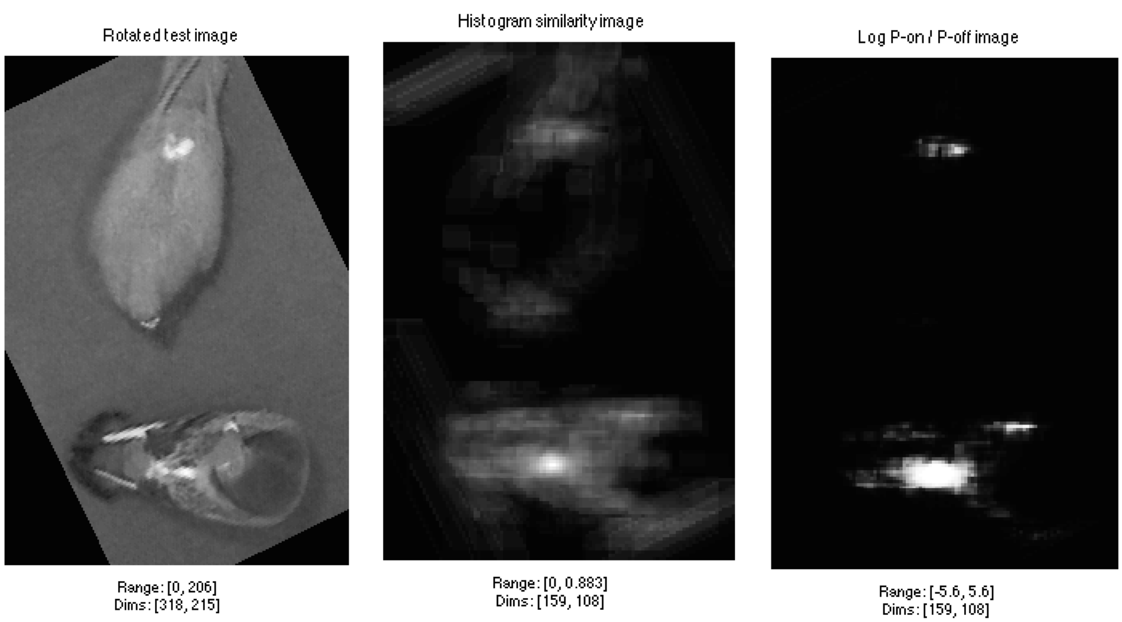
Results of the HOG training are shown in Figure 2 for tracking the targets of interest. From Figure 2a, the likelihood is strongest when the rotation of the image aligns with the same direction as the target. When the rotation of the image is not aligned with the target pose, the likelihood response is weaker, as shown in Figure 2b. Figures 2c-2d shows the HOG performance with 2 pigeons in the scene while tracking the grey pigeon's back. From observation, the global maximum of the log likelihood coincides with the target, but there are other local maxima. The true target may not always be the global maxmimum, such as in the case of tracking the red pigeon's head where the red pigeon's back is very similar, as shown in Figure 2e.



(a) Tracking grey pigeon's back (aligned)

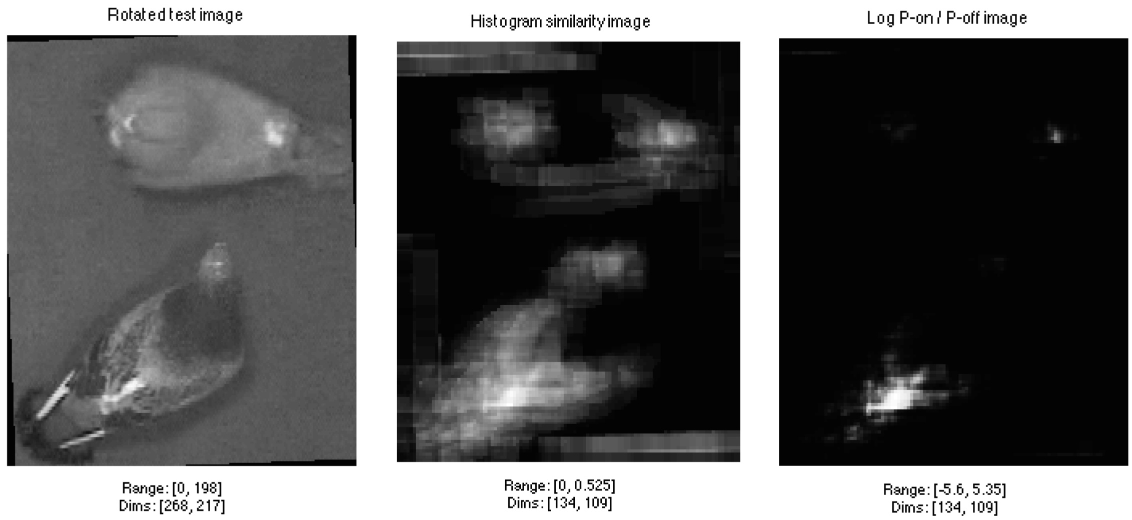


(b) Tracking grey pigeon's back (not aligned)

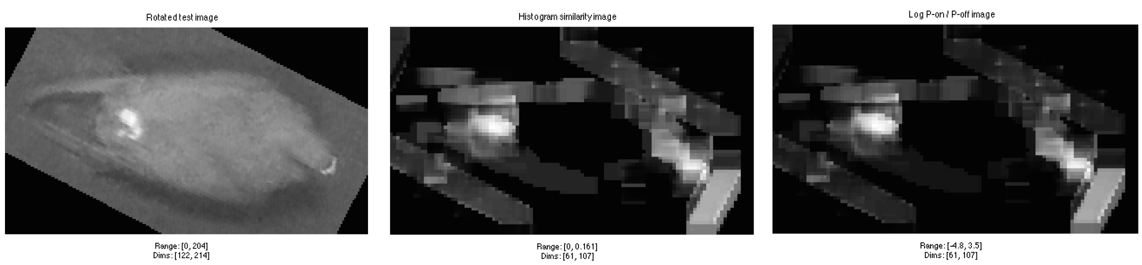


(c) Tracking grey pigeon's back (2 pigeons, aligned)

Figure 2: HOG test



(d) Tracking grey pigeon's back (2 pigeons, not aligned)



(e) Tracking red pigeon's head

Figure 2: HOG test

2 Particle Filter

The dynamics model uses a zero velocity random walk. The pose $\vec{a}_t$ can be generated from normal distribution $\sim \mathcal{N}(\vec{a}_{t-1}^{(j)}, C_D)$, where C_D is the covariance matrix and j is the sample number. The weights, $w_t^{(j)}$, of the particle set are normalized with the sum of all likelihood and the effective number of particles, N, is calculated.

$$w_t^{(j)} = \frac{p(z_t|a_t^{(j)})}{\sum_j p(z_t|a_t^{(j)})} \tag{1}$$

$$N = \frac{1}{\sum_j (w^{(j)})^2} \tag{2}$$

After modeling the dynamics and updating the weights of the particle filter, tracking of the targets of interests is achieved. The covariances for the dynamics are adjusted according to how fast the target is moving. Slower targets, such as the pigeons' backs, have smaller covariances than faster targets, such as the pigeons' heads. From observing the pigeons' movements, the head tends to move faster in translation and rotation (e.g. heads rotate and move forward and backward during walking) than the body. Thus, higher $\sigma_x = \sigma_y$, and σ_θ are selected for the head.

The covariances are determined through trial and error to find the best performance for tracking. Selecting covariances that are too high results in a large range of sampling, which can catch the target but also risk catching other similar HOG outlier patches. On the other hand, selecting covariances that are too small results in losing tracking of the target when the target is moving quickly. Through trial and error, covariances are selected for each target of interests that best balances between the points mentioned above. Figures 3-8 show some tracking results of targets through frames.

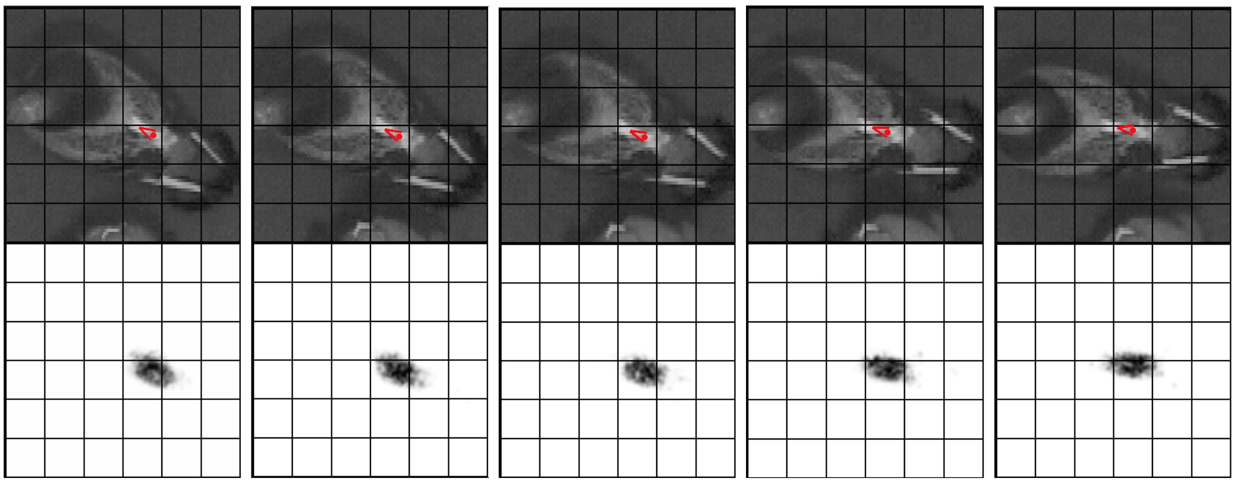


Figure 3: Tracking grey pigeon's back through frames

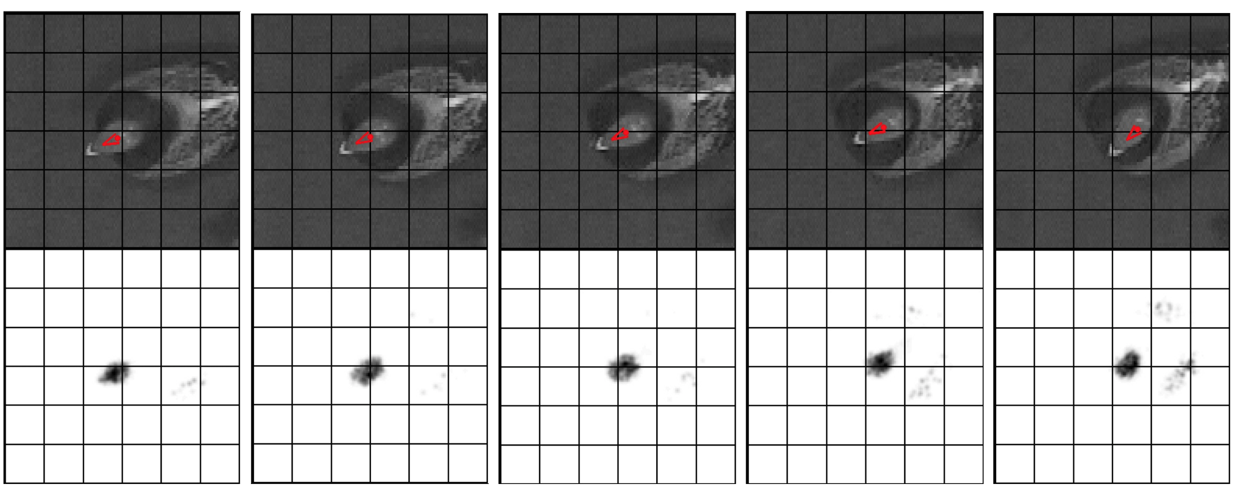


Figure 4: Tracking grey pigeon's head through frames

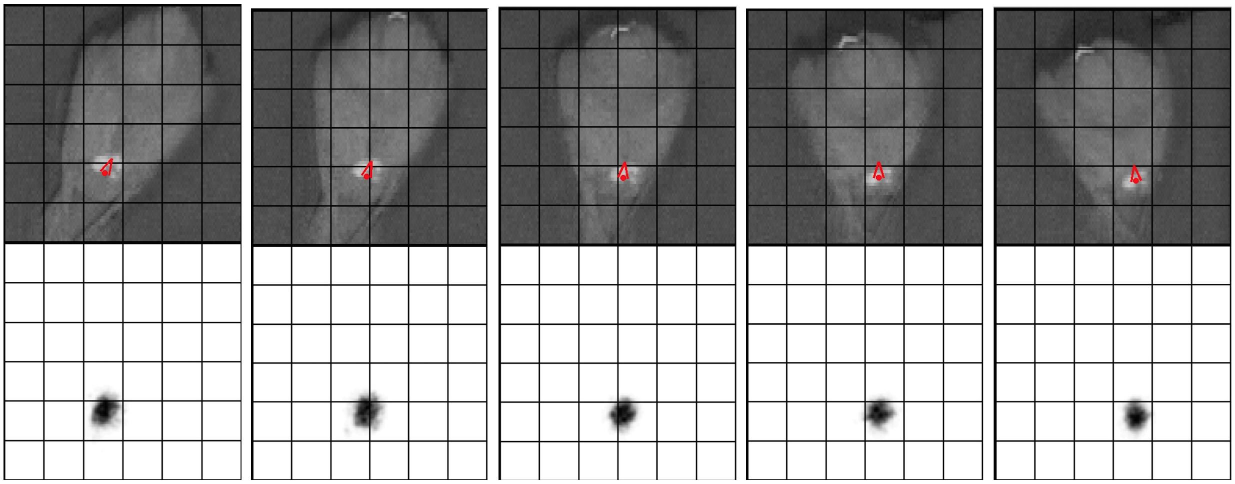


Figure 5: Tracking red pigeon's head through frames

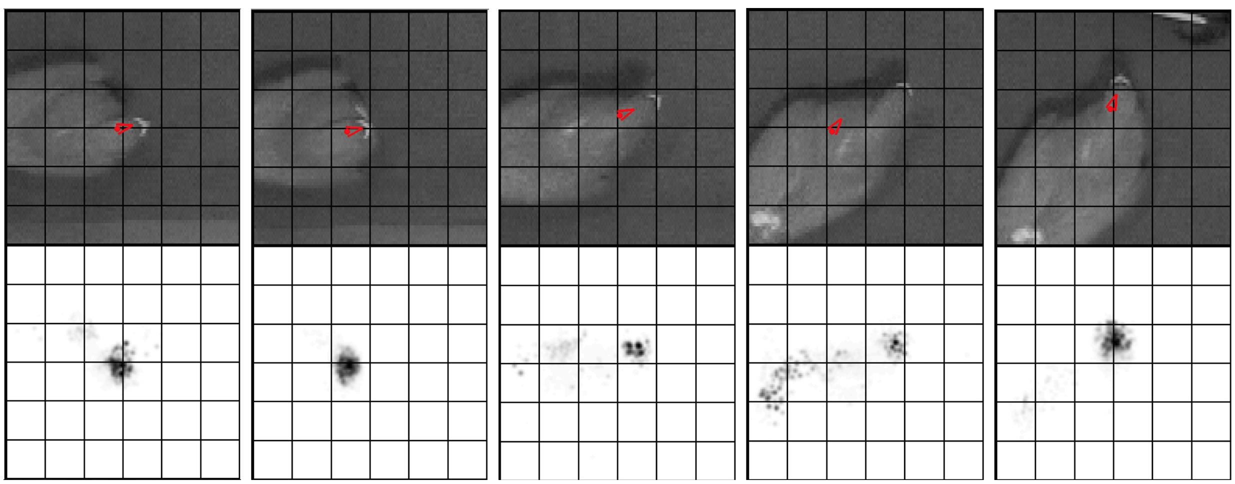


Figure 6: Tracking red pigeon's head through frames

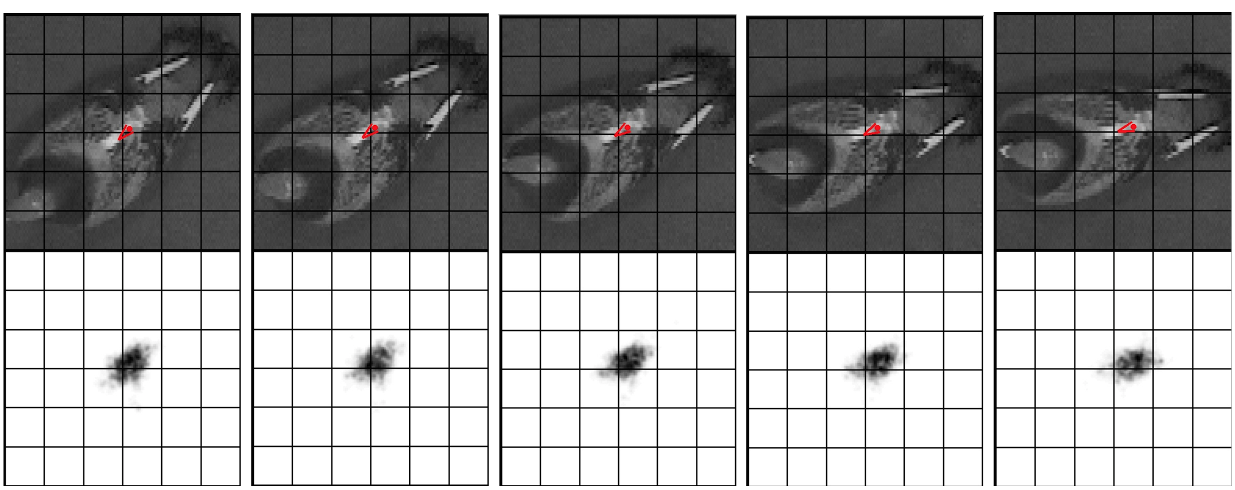


Figure 7: Tracking grey pigeon's (clone) back through frames

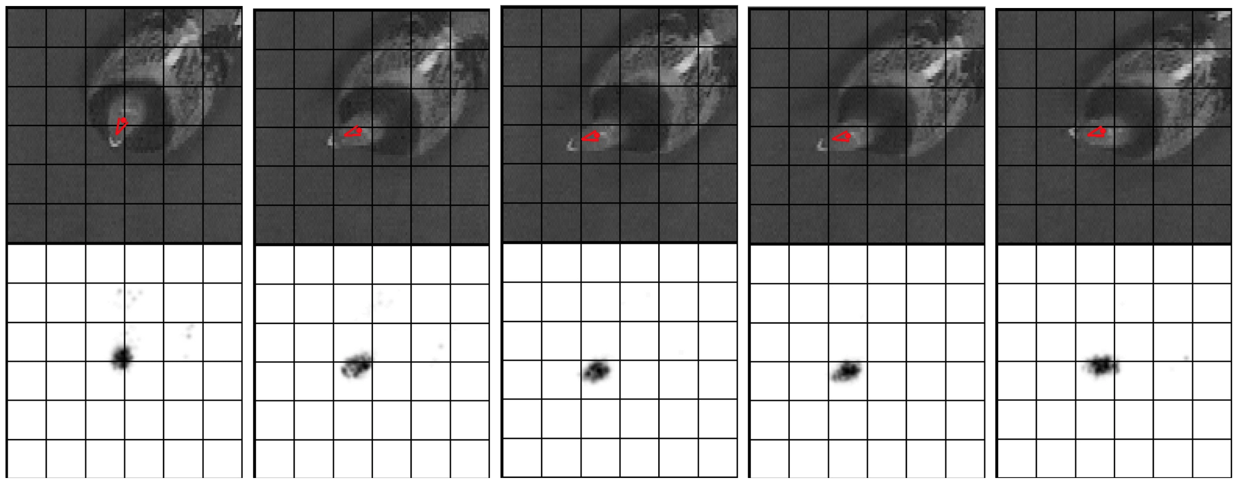


Figure 8: Tracking grey pigeon's (clone) head through frames

2.1 Evaluation of Results

The algorithm performs fairly well in tracking the targets in the given video. As shown in Figures 3-8, the tracker correctly estimates the poses of the targets through frames. However, there are cases where the tracker fails due to HOG similarities in a few frames, but the tracker eventually correctly tracks the targets again, as shown in Figure 6. The clusters of effective particles for the red pigeon, especially the red pigeon's head, is also less dense than what it is for the grey pigeons. Thus, tracking is more difficult and the algorithm sometimes tend to misrecognize another part of the pigeon's body for the head.

In general, there are a few requirements of the image sequence and target for there to be a reasonable expectation that the algorithm will work.

1. The target should have a distinctive visual feature, such as texture and colour, to distinguish itself from its neighbours. More specifically, the p_{On} and p_{Off} curves should have a distinguish separation. Since the estimated pose is based on all weighted particles, any false positive regions will skew the the results.

2. Movement speed of the target should also be consistent and smooth within small range of speeds. Otherwise, a single covariance matrix is not able to capture all movement sample. The head of the pigeons tend to accelerate and decelerate during certain points of the image sequence. Tracking these points are much more difficult than when the pigeon's head is staying still.

The particle algorithm's strengths lie in its ability to track targets with high dimensional non-linear non-Gaussian models by using sampling approximation. It is also able to converge on the target again even when the target is lost in an image sequence within reasonable limits. The algorithm's weaknesses lies within the simplification of the model. The particle filter is based on first order Markov model and the conditional independence of observations ignore some dependence in sequential data. Moreover, the observation history also affects the performance of the algorithm. The algorithm may converge on a false positive target and may lose the original target entirely. Another weakness is the algorithm's dependence on its initial guess. If the initial guess is not reasonable, the target may never be tracked. Finally, the algorithm fails in higher dimensionality, since the number of samples grow exponentially with state dimension.

There are a few failure modes of the algorithm. Under the conditions where there are similar texture around the target and the target moves quickly, the tracker may skew away from the target and the target may be lost. Once the target is lost, it may be hard to converge back on the target depending on the covariance parameters. Large covariances will give a better chance at relocating the target, but this will also impact the performance of the particle filter. As mentioned above, another failure mode is when the initial guess is poorly estimated. The object will not be tracked initially and the tracker may never converge onto the target.

3 Tracking Two Modes

Figures 9-10 show the simultaneous tracking on the two grey pigeons. At first, the algorithm identifies particle cluster of the two targets well. However, the algorithm loses the targets after many sequential frames and converges into one of the targets. For tracking grey pigeons' backs as shown in Figure 9, the cluster eventually converges to the back of the left grey pigeon. The effective particles for the left pigeon essentially has more weight determined by the movements of the two pigeons throughout the frames, and hence, more samples are drawn from that region. From observation, the left pigeon is staying relatively still compared to the right pigeon. The number of effective particles reduces sharply for the moving pigeon and the particles shift from the right grey pigeon to the left grey pigeon. For tracking grey pigeons' head as shown in Figure 10, the right pigeon's head initially has a denser and more intensive particle cluster associated with it compared to the left pigeon's head. With this weight bias, within the next few frames, the particle filter converges to the right target as the weights shift to the right cluster.

From these observation, an additional failure mode is present when tracking multiple targets. Tracking 2 modes with this algorithm is highly dependent on the relative velocity of the targets and is not feasible. It will track well initially, but as the weights of the particles shift around, it will evenutally converge to tracking one of the two modes. The number of particles reduces immediately for moving targets. During resampling, clusters with less samples have less weight. Eventually the clusters will disappear and it will be impossible to track them again. Furthermore, the likelihood function used for tracking two modes is trained with only one target in mind. In conclusion, using a single particle filter to track multiple targets is not feasible.

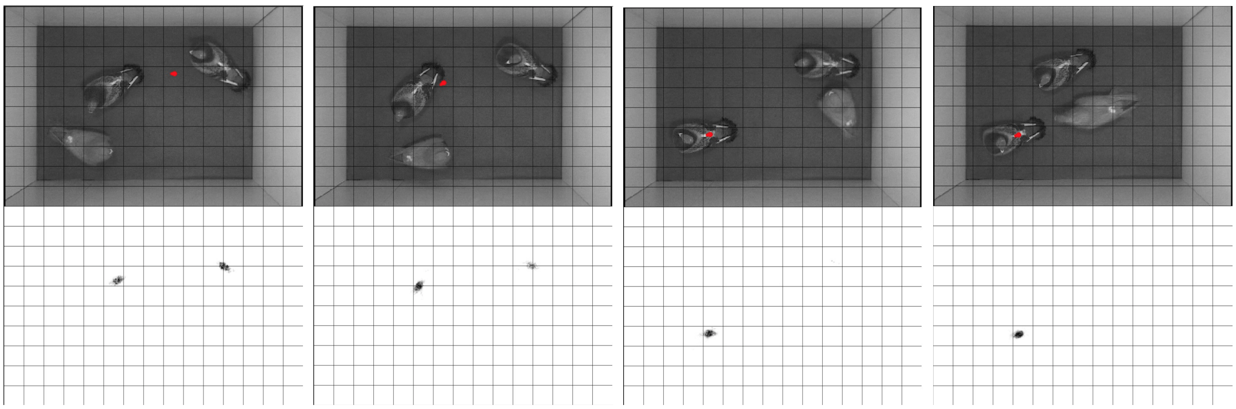


Figure 9: Tracking grey pigeon's back with two modes

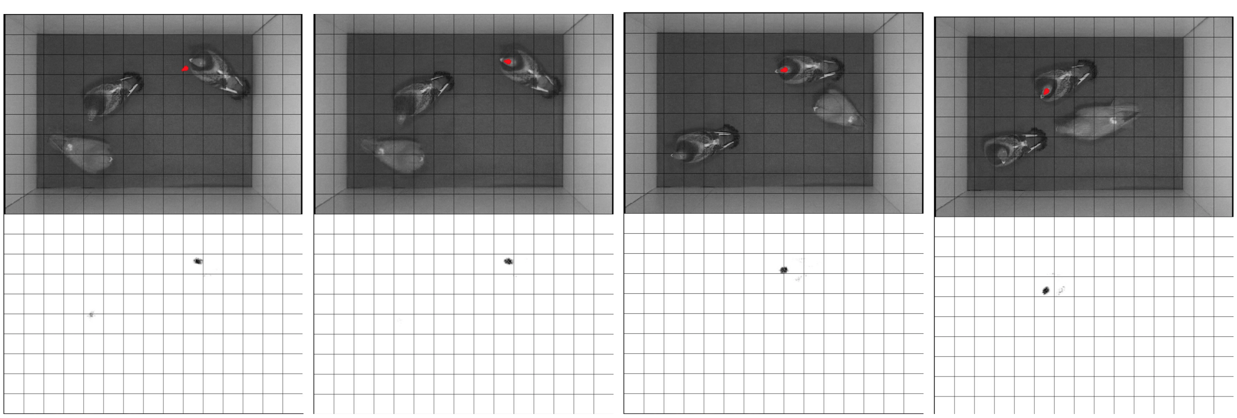


Figure 10: Tracking grey pigeon's head with two modes