CV Tracking Lab Report

Luca Sichi 19-928.100

Color_histogram.py:

This function implements a standard color histogram, that can adapt to the number of bins in each dimension. The hard part was to handle the fact that we only consider a frame and thus need to handle the border region.

Propagate.py:

In this file I implement the matrix A, and the propagation function. The matrix A depends on the motion model we use: if we assume constant velocity the matrix is 4x4, if we assume any motion model the matrix is 2x2. We sample “noise” according to sigma_position and/or sigma_velocity and then calculate our new particles accordingly.

Observe.py:

In this file we calculate the histogram for each particle, and then calculate its chi2 cost with the given function. We then normalize this by the sigma_observe and return the calculated weights.

Estimate.py:

This function just calculates the mean state by multiplying particle with its corresponding weight and then takes the sum over this, since our weights already contain a normalization factor.

Resample.py:

In this function we use np.random.choice to do the heavy lifting. We just take the returned indices and use them.

Video1.avi:

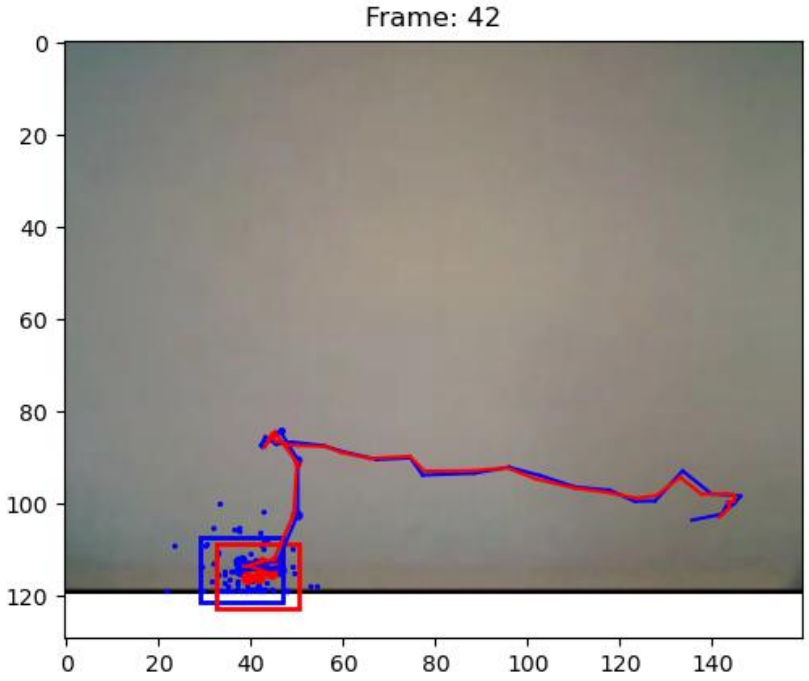
Here I chose the following parameters:

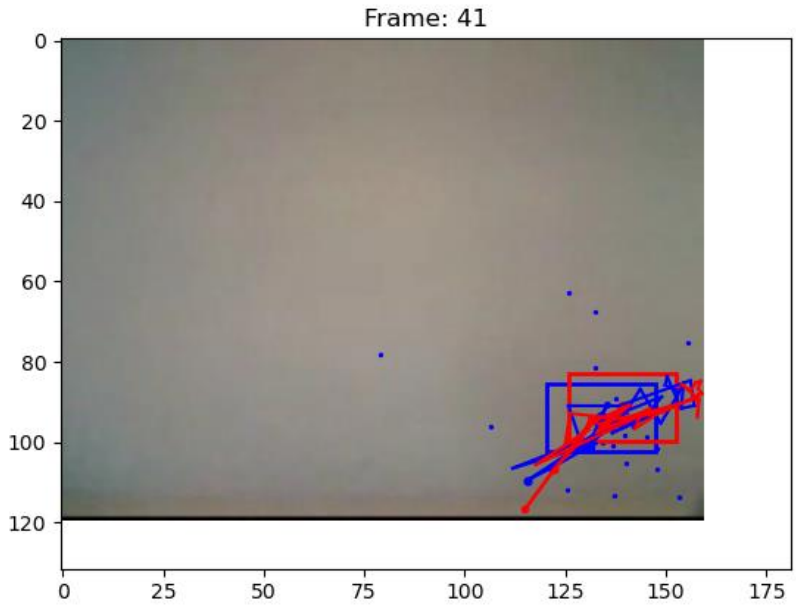
```
      "draw_plots": 1,
        "hist_bin": 16,
        "alpha": 0.4,
        "sigma_observe": 0.1,
        "model": 0,
        "num_particles": 80,
        "sigma_position": 5,
        "sigma_velocity": 5,
        "initial_velocity": (1, 10)
```

I did this by experimenting.

I noticed that the color histogram changes over time(the hand is illuminated differently) this meant that I had to adapt the histogram, which explains the alpha of 0.4

Below you can see that it works well with my parameters, and then a set of parameters for which it doesn’t work:





Video2.avi

For the next video I chose the following parameters by experimenting:

```
    "draw_plots": 1,
      "hist_bin": 12,
      "alpha": 0.1,
      "sigma_observe": 0.1,
      "model": 0,
      "num_particles": 60,
      "sigma_position": 15,
      "sigma_velocity": 1,
      "initial_velocity": (1, 10)
```

Here we decrease the alpha value, num particles and increase sigma position.

Q: What is the effect of using a constant velocity motion model?

A: We assume constant velocity of the object: meaning that if the object changes its velocity (change in direction may also change its velocity) we get very poor results, we can see this in video3.

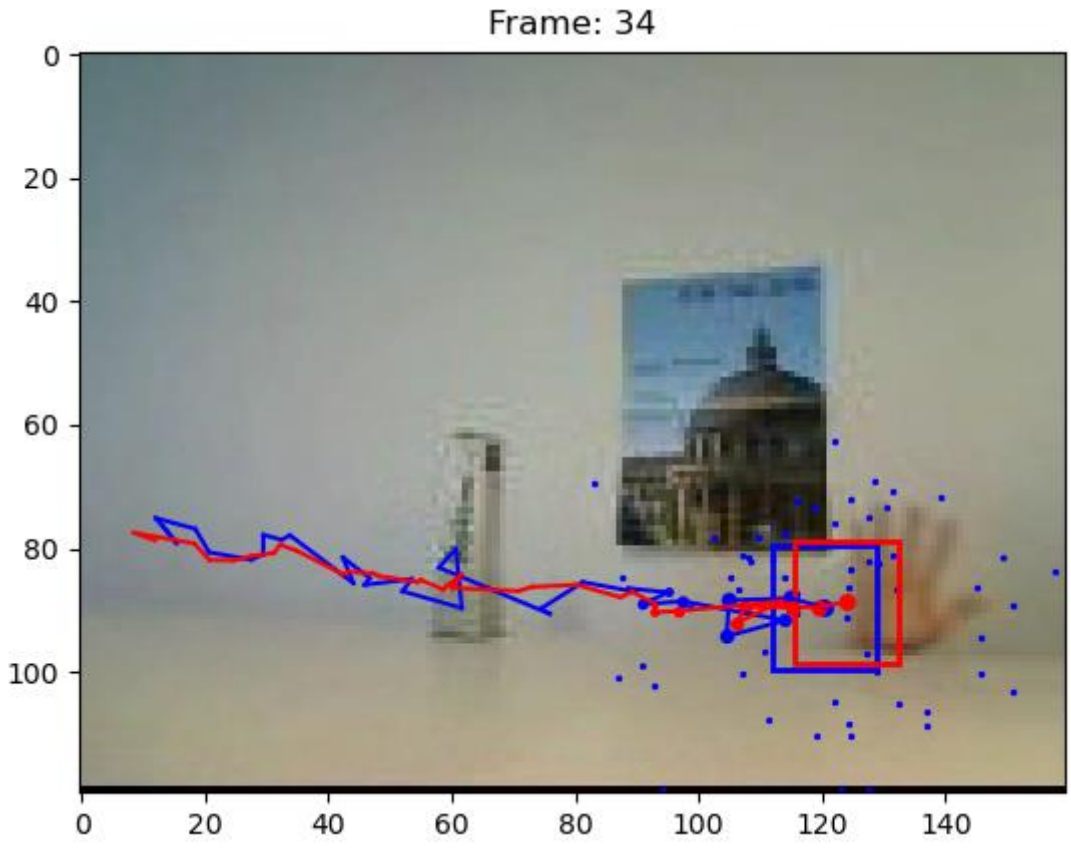
Q: What is the effect of assuming decreased/increased system noise?

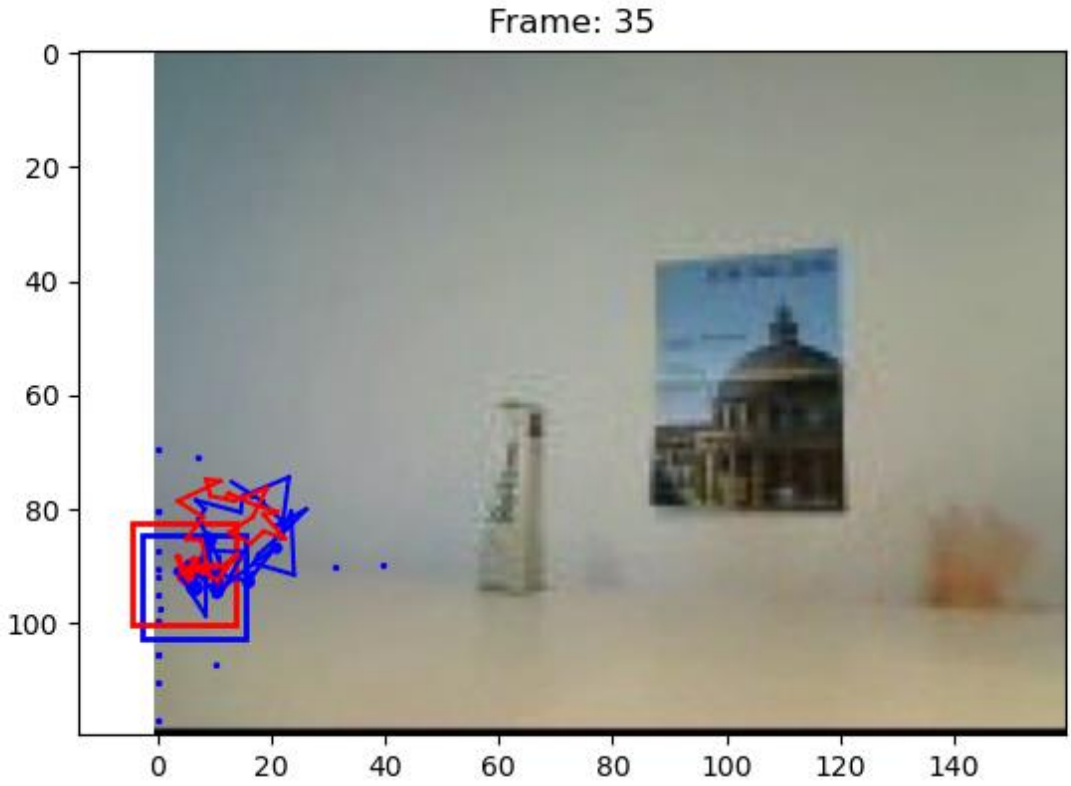
A: The effect of increasing the noise is that we “look” in further distanced positions for the object.

Q: What is the effect of assuming decreased/increased measurement noise?

A: increasing measurement noise means that the samples aren’t that close to each other, which may lead to a failure of tracking the object.

Below you can see the successful tracking of the hand with the given parameters, and an unsuccessful tracking:





Video3.avi

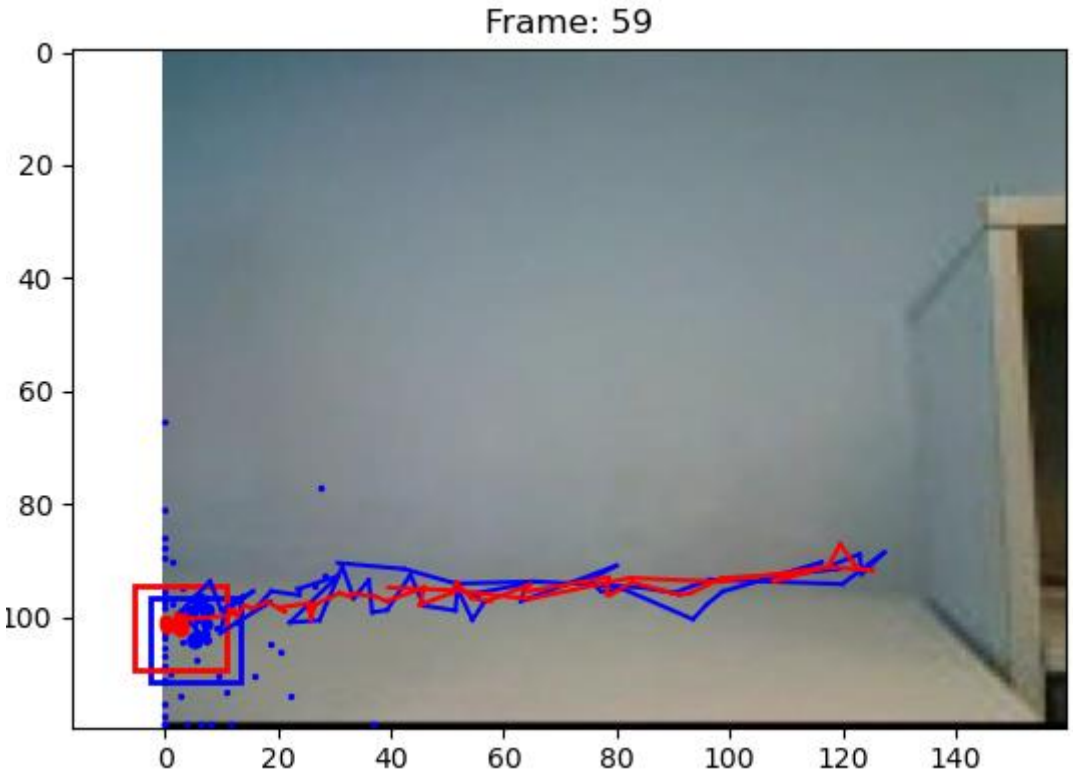
Used parameters:

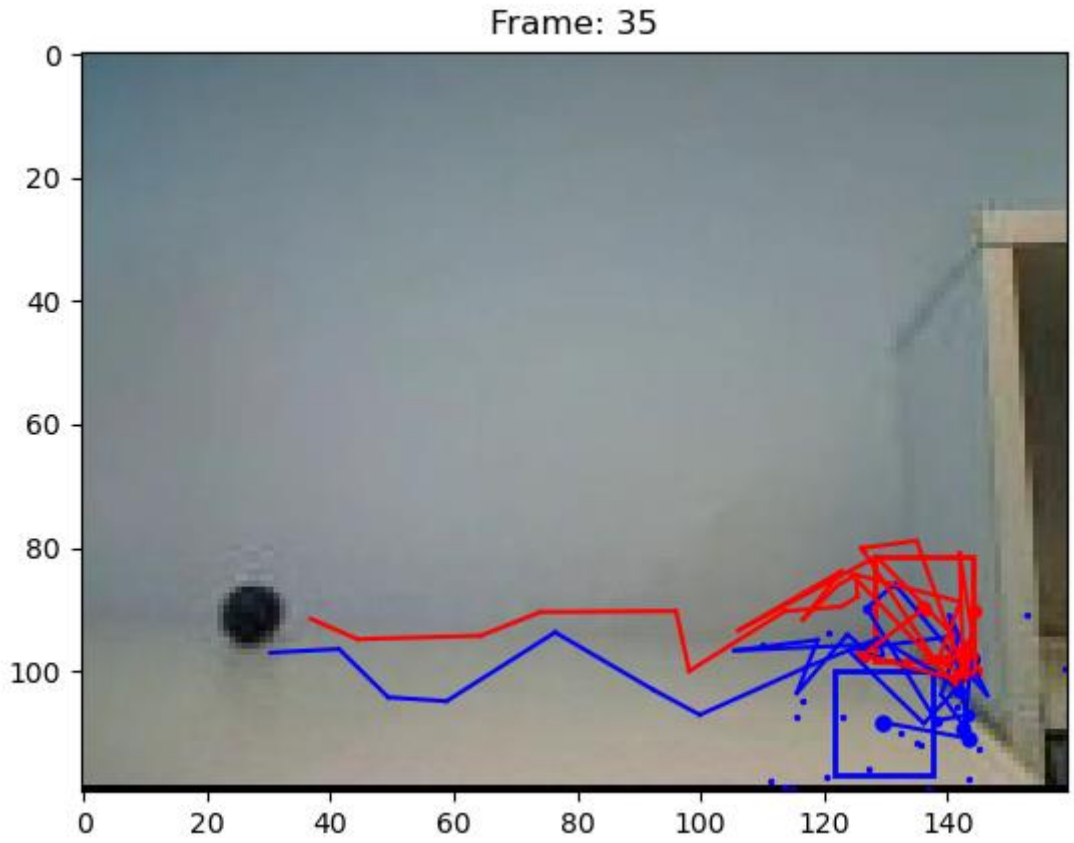
```
      "draw_plots": 1,
        "hist_bin": 16,
        "alpha": 0.1,
        "sigma_observe": 0.1,
        "model": 0,
        "num_particles": 50,
        "sigma_position": 15,
        "sigma_velocity": 1,
        "initial_velocity": (1, 10)
```

Since I used almost the same parameters here we can give the same answers as above.

However in this video the use of a constant velocity model would result in a very bad tracking, since the algorithm would loose the ball when it hits the wall.

Below you can see the result of tracking the ball with the described parameters and an unsuccessful tracking:





General remarks:

I noticed a big variance in the result depending on the exact selection of the initial frame.

Q: What is the effect of using more or fewer particles?

A: We look at more “possible” positions, which means we could get a better result at the cost of more computing time.

Q: What is the effect of using more or fewer bins in the histogram color model?

A: This parameter affect’s how well we approximate the real color, for example if we choose a very low number of bins we can only differentiate between shades of colors. This might be desirable if we have small deviations in the color of the object, since it would still yield the same histogram.

Q: What is the advantage/disadvantage of allowing appearance model updating?

A: If for example the object gets illuminated more and more, we can adapt our algorithm to allow for this change by allowing the update. On the other hand if we ever loose track of our object it may be very difficult to recover from this.