

Mean-Shift tracking

Marko Medved

I. INTRODUCTION

In this assignment, the mean shift method was implemented and used to develop a tracker, which was evaluated on the VOT14 dataset. The model's performance was analyzed under different parameter settings. The tracker was further enhanced by incorporating additional weights into the histogram, with the weights adjusted based on the background characteristics of the tracked object.

II. EXPERIMENTS

A. Mean shift mode seeking on the given example

The Mean Shift method was tested with various parameters to evaluate its performance. The kernel size, which defines the region for the calculation of the current step, was adjusted. Larger kernels led to faster convergence, but excessively large or small kernels prevented convergence to the local maximum. Different starting positions were tested, showing that starting points in low-probability areas prevented movement, and for multi-modal functions, the starting point determined which maximum the algorithm would reach.

Three convergence criteria were evaluated: (1) step size, stopping when movement is under one pixel; (2) Euclidean distance, halting when consecutive positions differ by less than two units; and (3) change in probability, stopping when the probability shift is minimal. The sub-pixel step size was the most consistent but slowest, while the probability-based criterion sometimes caused premature convergence if early changes were small. Figure 1 shows how these parameters influenced convergence. Note that each color represents a different starting position.

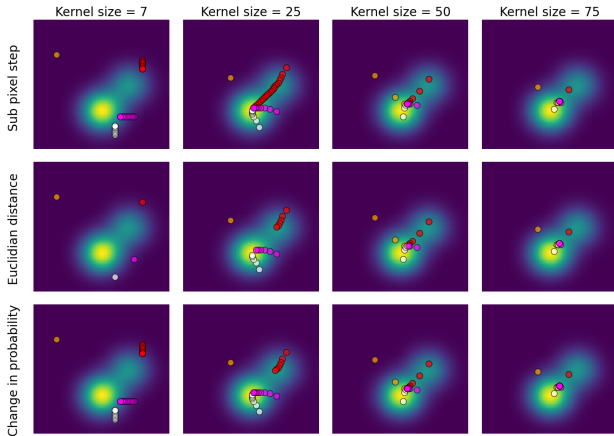


Figure 1. Comparison of the mean shift method with different kernel sizes, convergence criteria and starting positions

B. Mean shift mode seeking on custom examples

The algorithm was then tested on three custom functions, shown in Figure 2. In these tests, the sub-pixel convergence criterion was used, and the kernel size was set to 25. For each function, four different starting positions were evaluated. The

first function is a Gaussian distribution, where the algorithm consistently converges to the global maximum, provided the starting point is not in a region with near-zero probability. The second example is the Laplacian of Gaussian (the Mexican hat function), which has multiple local maxima. Here, the convergence outcome strongly depends on the starting position. The final example is a small section of the Julian Alps (around 46° North and 14° East). Due to the presence of many local maxima, the starting position determines which peak the algorithm will ascend to.

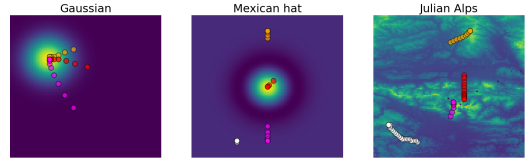


Figure 2. Comparison of the mean shift method on different functions

C. Basic tracker implementation

The tracker was implemented and tested on the entire VOT 2014 dataset. The number of failures and the tracking speed (measured on my personal laptop) are summarized in Table I. Overall, the tracker failed a total of 33 times across the dataset. While the tracker performs reliably in most cases, it struggles more with certain sequences, such as hand2, fish1, torus, and tunnel. The reasons behind these failures will be explored in detail in the next section.

Table I
BASIC TRACKER RESULTS

| Sequence | Failures | Speed | Sequence | Failures | Speed |
|------------|----------|----------|-----------|----------|----------|
| ball | 1 | 1823 FPS | david | 1 | 1042 FPS |
| basketball | 0 | 510 FPS | diving | 0 | 1328 FPS |
| bicycle | 1 | 1047 FPS | drunk | 1 | 700 FPS |
| bolt | 2 | 1425 FPS | fernando | 2 | 358 FPS |
| car | 0 | 2017 FPS | fish1 | 3 | 1571 FPS |
| fish2 | 1 | 1224 FPS | motocross | 2 | 504 FPS |
| gymnastics | 0 | 1667 FPS | polarbear | 0 | 2020 FPS |
| hand1 | 2 | 923 FPS | skating | 1 | 1098 FPS |
| hand2 | 5 | 2610 FPS | sphere | 0 | 1285 FPS |
| jogging | 1 | 1704 FPS | sunshade | 0 | 928 FPS |
| surfing | 0 | 3179 FPS | torus | 3 | 1170 FPS |
| trellis | 2 | 1449 FPS | tunnel | 4 | 2684 FPS |
| woman | 1 | 2525 FPS | | | |

D. Failure cases discussion

Figure 3 shows the cases where the tracker experienced the most failures. In the fish1 and hand2 sequences, failures occurred due to similar color distributions between the tracked and ground truth bounding boxes, making it hard to detect movement. In the torus and tunnel sequences, scale changes added complexity: in torus, object rotation altered the color distribution, and in tunnel, the motorbike's decreasing size led to an inaccurate bounding box.

These failure cases suggest two potential improvements: addressing scale changes by testing different template scales and minimizing the influence of the background. The latter is discussed in the final section.

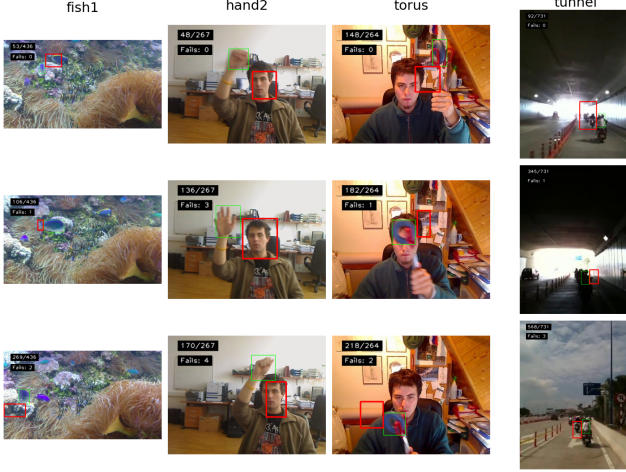


Figure 3. Mean shift tracker failure cases

E. Parameter tuning

Four key parameters were tested: σ (kernel size), histogram bin count, iterations per frame, and α (template update rate). The algorithm was evaluated for all combinations of 4 bin counts, 4 kernel sizes, 3 iteration values, and 4 update rates.

To determine the optimal number of iterations and bins, results were aggregated based on these parameters (Figure 4). The data shows that using only one iteration significantly reduces performance, while 10 and 20 iterations yield similar results, with average FPS of 1482 and 1484, respectively. This indicates negligible impact on speed with either 10 or 20 iterations.

Regarding the number of bins, 8 or 16 bins deliver the best performance. While 8 bins slightly underperform compared to 16, they offer a notable increase in FPS. On a personal laptop, 16 bins reached an average FPS of 2151, making it the preferred choice unless running on a device with limited computational power. Based on that 20 iterations and 16 bins were used for further evaluation.

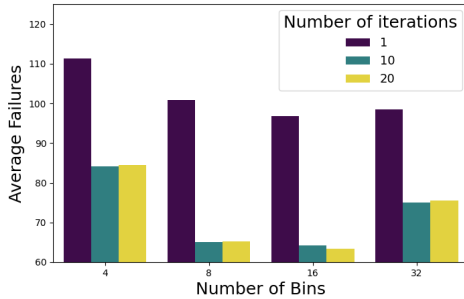


Figure 4. Performance of the means shift tracker with different numbers of bins and numbers of iterations

To determine the best values for σ and α , we refer to Figure 5. Since FPS remains consistent across parameter values, the

focus is on minimizing failures. The optimal σ is clear (with $\sigma = 1$ we can reach as low as 29 failures), but the choice of α depends on the problem. For targets whose appearance changes frequently, $\alpha = 0.01$ is preferred, while for stable targets, $\alpha = 0$ is more suitable.

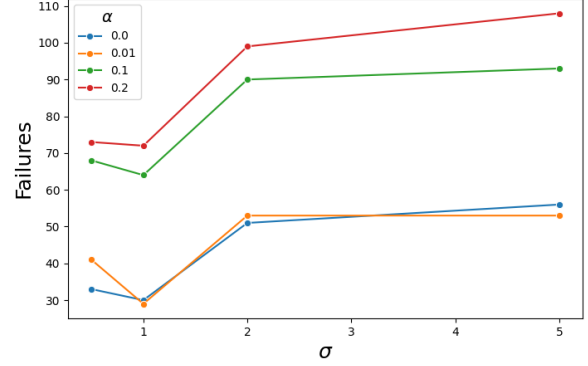


Figure 5. Performance of the means shift tracker with different kernel sizes and update rate parameters

F. Feature selection by accounting for the background in different color spaces

To mitigate the influence of the background, the method applied smaller weights to the regions in the histogram where the background color predominated. The tracker was also evaluated using five different color spaces. Additionally, a new parameter was introduced: the size of the background area. Since the optimal value for α was not definitive in the previous evaluation, two promising values ($\alpha = 0$ and $\alpha = 0.01$) were considered in this experiment.

The results showed that the optimal parameters are $\alpha = 0$ and a background size twice the size of the template. The template region itself was excluded when computing the background contribution.

Table II presents the number of failures across the entire dataset for each color space using these optimized parameters, showing a clear improvement over the baseline method. Note that, the performance on the most challenging videos (hand2, fish1, torus, and tunnel) improved, with the number of failures reduced to 4, 1, 0, and 3, respectively.

Table II
PERFORMANCE OF THE IMPROVED TRACKER ACROSS DIFFERENT COLOR SPACES.

| Color space | Failures | Average Speed [FPS] |
|-------------|----------|---------------------|
| RGB | 23 | 981 |
| LAB | 26 | 831 |
| HSV | 21 | 1002 |
| YCrCb | 25 | 1230 |
| BGR | 23 | 1646 |

III. CONCLUSION

In this assignment, the mean shift algorithm and tracker were implemented. The analysis identified the problematic test sequences and explored the potential reasons for failure. The results demonstrated that the tracker's performance can be significantly improved by carefully tuning the parameters and incorporating background information into the model.