MEAN SHIFT ALGORITHM

Data Structures and Algorithms Term Paper-2020

Vikhyat Dhamija
Email ID: vikhyat.dhamija@rutgers.edu
RU ID:194003013
Net ID: vd283

Contents

Abstract

The purpose of this paper is to analyze the Mean Shift Algorithm , which is used as a non -parametric estimator of density gradient and is used in various applications like image segmentation , object tracking etc. This paper will provide with the intuitive understanding behind this algorithm , its detailed mathematical analysis and also the detailed analysis of its major application of Object Tracking. In the end ,the paper will conclude with the various issues related to mean shift technique , with sufficient points to convince that "Mean Shift" is quite a powerful technique/tool especially in the toolbox of Computer Vision.

1. Introduction:

In the formal terms we can state that the **Mean Shift Algorithm** is **an algorithm** of **finding modes** in a set of data samples, **manifesting an underlying probability density function (PDF) in RN.**

In simple terms, it can be understood that in a region of data points i.e. the spatial distribution of the data points , we create a small region of interest(ROI or filter or window) with its center and then calculate the mean (or some kind of weighted mean) of the data points with in that current position of ROI/window in the region of Data points and then that the center of ROI shift towards the mean (or can be said that the center of mass of that region).The vector from the current center of ROI towards the mean/weighted mean or the center of mass of the region of interest is called the **mean shift vector.** And proceeding in this manner , i.e. by shifting the center of ROI/window towards the mean of the ROI/window i.e. following the mean shift vector , we finally reach the region where the mean is equal to the center which leads to no more shifting. This final convergence point/center is of high significance because then our ROI is standing over/reached over/converged over the area/region in the data space region , which is **the densest region , which provides with quite a useful applications in the field of computer vision** . So , intuitively , in this algorithm we are continuously moving our small ROI to reach the densest area in the region of data points.

2. Elaboration of the Concept:

Let us look at the terms used in the formal definition , in order to understand the formal definition:

RN is the Region of Data Points (it can be imagined as the two-dimensional region of pixels in terms of images)

Probability density function (PDF):

Let us focus on the line in the definition :

"manifesting an underlying probability density function (PDF) "

We need to understand that what is meant by the **Probability Density Function**. Consider the Two-D space of our RN where PDF can be seen as a three-dimension function where t**he X-Y region is the region of the**

data points and the **Z Axis signifies the function value** that represents the **density of the data points in the region** . This function is called as **Probability Density Function(PDF)** because generally we see them in the **scale 0-1** as it signifies the probability of any random data point to be found at the particular position in the space.

<u>**Example to understand the above concept**</u>

Consider an example like India has 20 percent population , Europe has 5 percent population , United States has 10 percent population of the world so we can interpret the probability that any human's probability to be found in any of this country is this density of the population like **0.2** for India , **0.05** for Europe and **0.1** for United States.

As we saw, that the **mean shift algorithm** is stated to **manifest** this underlying **Probability Density function**. This can be visualized as , considering the simple description above that , in this algorithm moving our small ROI we are moving in the path of increasing probability i.e. moving from current area to denser area and stops when reach the densest area i.e. the peak in the **Probability Density Function.**

This property of the mean shift algorithm leads to its utilization in many applications which is called as <u>**Non-parametric Density GRADIENT Estimation (Mean Shift)**</u>

It is also important to mention that it is called as **Non-Parametric** because **parametric PDFs are those which have a common/general shape like the gaussian which can be represented only by the mean and the variance like parameters but the Non-parametric PDF is the one calculated over the whole data i.e. like in our case by using the mean shift algorithm over the data and not just can be described by the few parameters like mean , variance of the Data .**

3. Description and Analysis:

Using the ideas above for the Mean Shift Algorithm , let us analyze it:

$$\boldsymbol{M = \frac{1}{n} * \sum xi - y_0 \; where \; i \; varies \; from \; 1 \; to \; n}$$

M is the mean shift vector in the one dimension but as the y_0 can be a point in 2D and 3D so accordingly the mean shift vector is calculated.

As we said generally, we take the weighted mean of the points which can be represented as :

$$\boldsymbol{Mw = \frac{\sum_{i=1}^{n} wi(y_0) * xi}{\sum_{i=1}^{n} wi(y_0))} - \; y_0}$$

Mw is the weighted mean by which we assign some weight to the points and take the weighted mean to calculate the mean shift (or mean shift vector in case of multi dimensions)

Important Point: *The mean shift vector has the direction corresponding to the gradient of the PDF and computing iteratively it leads to the peak of the PDF.*

The above is the simple idea , now let us see how we derive the PDF, using Kernel Density Estimation.

$$\boldsymbol{P(x) = \frac{1}{n} * \sum_{i=1}^{i=n} K(x - xi)}$$

Here **K is called as a Kernel Function.**

Some of the properties of the Kernel Function are :

a. Normalized Function = $\int k(x)\, dx = 1$ over the region
b. Symmetric Function= $\int xk(x)\, dx = 0$

Kernel function can be seen as the function which maps the distance of any point (like x above) from the other points(xi above) i.e. provide weight to the distance of the point x with other points xi , which provide us with the Probability density at a Point x.

Hence , the process can be seen as superposition of kernels, centered at each data point is equivalent to convolving the data points with the kernel.

Various types of Kernel Functions are there for this convolution:-

a. **Epanechnikov Kernel :**

$$\boldsymbol{= c(1 - \left(mod(x)\right) \quad) \; for \; mod(x) \leq 1 \; and \; 0 \; elsewhere}$$

b. **Uniform Kernel:**

$$\boldsymbol{= c \; for \; mod(x) \leq 1 \; and \; 0 \; elsewhere}$$

c. **Normal Kernel:**

$$\boldsymbol{= c * e^{(-(mod(x))^2)} for \; mod(x) \leq 1 \; and \; 0 \; elsewhere}$$

Note that:

a. Mean shift algorithm in the computer vision application uses generally **Epanechnikov Kernel** as the kernel function.
b. Mod x above represents the distance from the origin i.e. 0

Let us see mathematically that how the Mean shift represents the gradient of this Probability Density function.

$$P(x) = \frac{1}{n} * \sum_{i=1}^{i=n} K(mod(x - xi)^2)$$

$$\nabla P(x) = \frac{1}{n} * \sum_{i=1}^{i=n} \nabla K(mod(x - xi)^2)$$

$$\nabla P(x) = \frac{1}{n} * 2 * \sum_{i=1}^{i=n} (xi - x) K'(mod(x - xi)^2)$$

$$\nabla P(x) = \frac{1}{n} * 2 * \sum_{i=1}^{i=n} xi * K'(mod(x - xi)^2) - \frac{1}{n} * 2 * \sum_{i=1}^{i=n} x * K'(mod(x - xi)^2)$$

$$\boxed{\nabla P(x) = \frac{2}{n} * (\sum_{i=1}^{i=n} K'(mod(x - xi)^2) * \left[\sum_{i=1}^{i=n} \frac{xi * K'(mod(x - xi)^2)}{K'(mod(x - xi)^2)} - x\right])}$$

Red colored part can be seen is simply the mean shift calculated at x using the k' (differentiation of K function) as the kernel function and we can observe that this mean shift is proportional to the $\nabla P(x)$

So this is the mathematical proof showing that by the means of calculating the mean shift we can calculate the gradient of PDF and hence using which we can create the underlying PDF.

4. Application:

Non-Rigid Object Tracking

The mean-shift algorithm is an efficient approach to tracking objects whose appearance is defined by histograms. The above point is very important to note that "***efficient approach to tracking objects whose appearance is defined by histograms***"

What are the objects that are defined by histograms? The answer is the **Non-Rigid Object** whose shapes do not remain fixed.(like a walking person), it is hard to specify an explicit 2D parametric motion model . Appearances of non-rigid objects can sometimes be best modeled with color distributions.

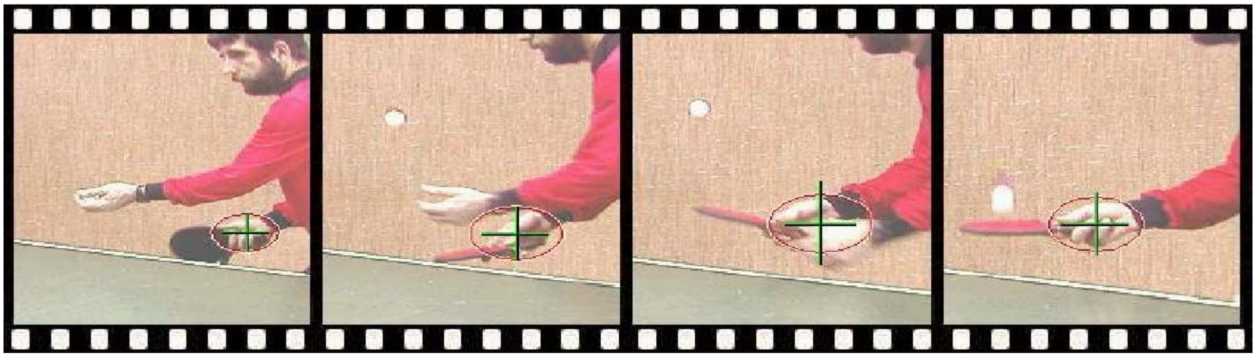


Figure 1.Source-Web

As in the above it can be seen that we are tracking the right-hand movement in various frames of the video. Let us see that how mean shift is used for this:

a. Target Representation

1. Choose a reference in the current frame (like the hand in the first frame)
2. Choose the feature space(like color, gray scales etc.)
3. Create a model of a reference (1) in chosen feature space(2)(Like color distributions in terms of histograms which is a PDF)
   Explain: Consider the Histogram where x axis represents the color and the height of the bars in a histogram represent the density of the color (can be visualized as a percentage/probability of the particular color among all)

So after this step , we have target representation is ready in terms of its **PDF.**

Then this histogram(PDF) is represented in terms of vector

$\overline{q}$ = { q1,q2,……………………..qm} q vector is a m dimension vector having the m values corresponding to m color bins as described above.

$\overline{p(y)}$ = { p1,p2,……………………..pm} p vector is a m dimension vector having the m values corresponding to m color bins. This p(y) represent the PDF at the position y which needs to be compared with the model target representation i.e. vector q

In order to compare the two histograms or vectors we use the Similarity Function:

Similarity Function : $\boldsymbol{f(y) = F(q, p(y))}$

So the position y where this similarity function f(y) will be maximum will be the tracking location of the Target Object to be tracked.

Bhattacharya Coefficient is used to find the similarity between the two :

Hence, Similarity function =$\sum_{u=1}^{m} \sqrt{q_u p_u(y)}$ **(Bhattacharya Function)**

This coefficient basically provides us with the Cosine value of angle between the two vectors and we have to maximize this function at a particular location y which will be the tracking location of the target object.

Now let us see how the $\boldsymbol{p_u(y)}$ is calculated :

$\boldsymbol{p_u(y)}$ **is the probability of the occurrence of color u where** $\boldsymbol{u} \in \{\, \mathbf{1}\ \boldsymbol{to}\ \boldsymbol{m}\}$ **i.e. the range of colors.**

For understanding this $p_u(y)$, visualize the region (like ROI/window in our description):

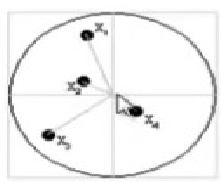


Figure 2:source web

So like the region as shown above , suppose that 4 points in this selected region has the same color , so similarly we will look individually for all the points of each color in u set and on these points we will assign the weight and then add them up to find the probability of occurrence of that particular color in that region.

So mathematically :

$$\boldsymbol{p_u(y) = C * \sum_{i=1}^{n} K(\|y - x_i\|^2)\ \delta(S(x_i) - u)} \ldots\ldots\ldots eq1$$

The meaning of this equation is that :

K (Kernel Function as described previously) is assigning weight to distance of y (i.e. the center) from all the points x_i but we have used this term $\boldsymbol{\delta(S(x_i - u))}$ because we have to take under consideration points of particular u in order to calculate the p_u i.e. the probability density of particular u. As the function(impulse function) $\delta(S(x_i) - u)$ = 1 when $S(x_i) = u$ that means for all the points xi for which color is u under consideration.

So goal now to track the target object is to look for the target object in the continuous frames of video and the tracked location is y where the similarity function is maximum.

Now let us see that how the mean shift is used for finding such similarity:

$$\varphi(y) = \sum_{u=1}^{m} \sqrt{q_u p_u(y)}$$

Taylor expansion of $\sum_{u=1}^{m} \sqrt{q_u p_u(y)}$ is:

$= \sum_{u=1}^{m} \sqrt{q_u \widehat{p_u}(y_0)} \; + \frac{1}{2} * \sum_{u=1}^{m} \widehat{p_u}(y) * \sqrt{q_u \div \hat{p}_u(y_0)}$ + other insignificant terms

= $\sum_{u=1}^{m} \sqrt{q_u \widehat{p_u}(y_0)} \; + \frac{1}{2} * \sum_{u=1}^{m} \widehat{p_u}(y) * \sqrt{q_u \div \hat{p}_u(y_0)}$

Now in the above equation we put the general probability density function $p_u(y)$ from the above eq 1 :

$$p_u(y) = C * \sum_{i=1}^{n} K(\|y - x_i\|^2)\, \delta(S(x_i) - u)$$

Then we get the similarity function as (note the red term is a constant) :

$= \frac{C}{2} * \sum_{i=1}^{n} [\, \sum_{u=1}^{m} [\sqrt{q_u \div \hat{p}_u(y_0)} \; * \delta(S(x_i) - u)] * K(\|y - x_i\|^2)]$

We need to understand this equation :

This equation of similarity function is very similar to the **PDF function at y** just for every pixel point we multiply the weight $w_i = \sqrt{q_u \div \hat{p}_u(y0)}$ term(depending on the color u of the pixel) to the term $K(\|y - x_i\|^2)]$ which is the weighted distance value for every data point or pixel from y in a window.

So above equation can be re written as :

$= \frac{C}{2} * \sum_{i=1}^{n} w_i * \; K(\|y - x_i\|^2)]$

Generally, we normalize the distance from y to x_i by dividing it by **h** i.e. the radius of the test circle(ROI/window/kernel radius) whose center is y .

So above equation can be re written as :

$$= \frac{C}{2} * \sum_{i=1}^{n} w_i * K\left(\frac{\|y-x_i\|^2}{h}\right)$$

The mode of this above function is the sought maximum.

And the maximum peak of this PDF can be found out using the **Mean Shift algorithm/process** as we have already concluded that **mean shift algorithm** provides the gradient of the **P(y)** function , so after following the mean shift algorithm we reach the point where there is the peak of the PDF and hence in the above case we will follow the mean shift process to reach the y where the Similarity Value(Similarity Chances as the similarity function is underlying PDF) are maximum.

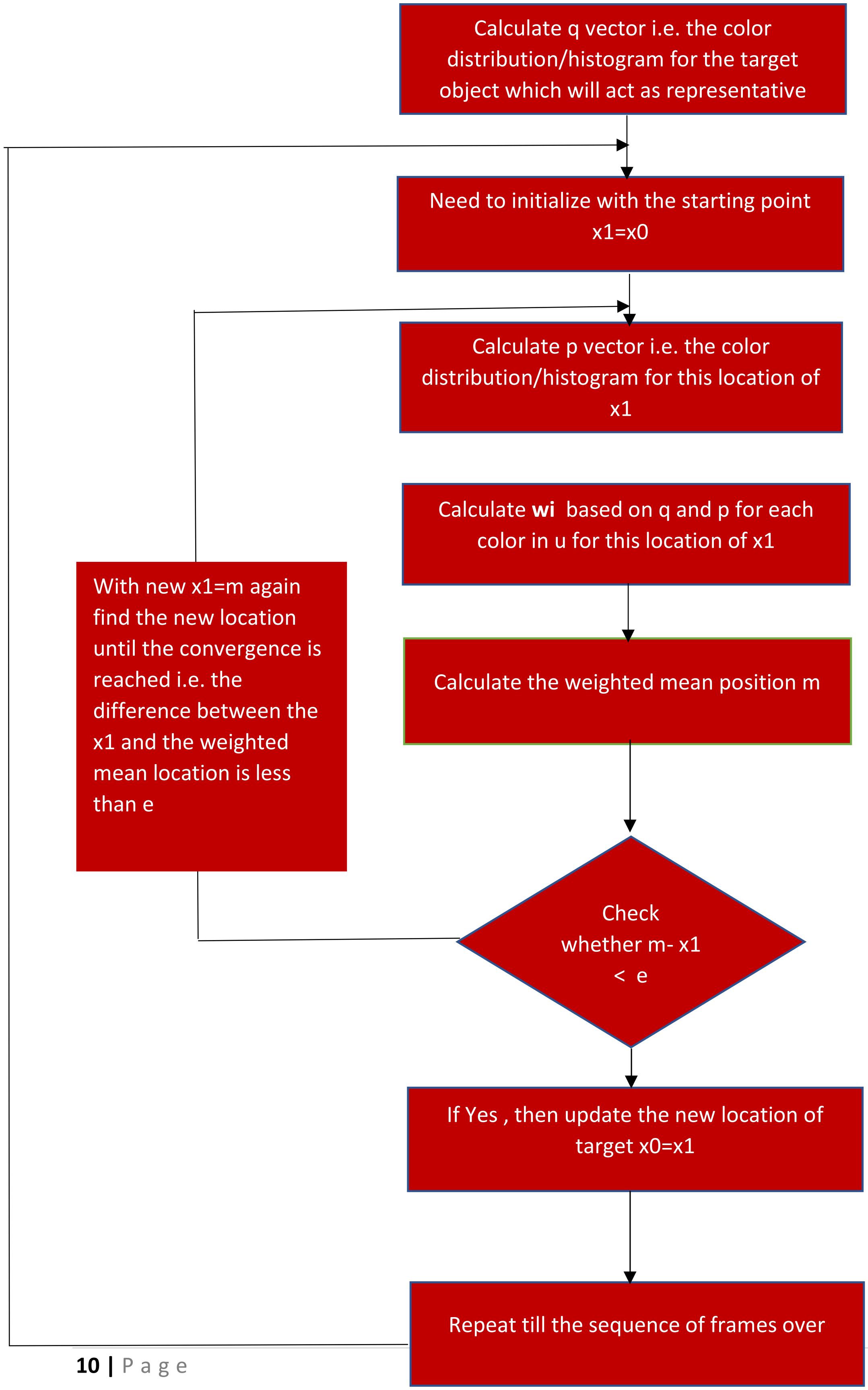
Mean shift Vector =

$$\boxed{\sum_{i=1}^{i=n} \frac{xi * wi * K'\left(\left[\frac{mod(y-xi)}{h}\right]^2\right)}{wi * K'\left(\left|\frac{mod(y-xi)}{h}\right|^2\right)} - y}$$

<u>Final Point of convergence of y will give the location of the target object.</u>

Notes: Generally, Epanechnikov kernel is used as the kernel function where K'(x) is the Uniform Kernel function.

Algorithm for the Object Tracking Application based on Mean Shift algorithm :



5. Conclusion/Summary:

In this paper , we have analyzed the basic concepts of mean shift algorithm , its mathematical analysis and one of its most important application i.e. the tracking of an object in a very simple and exhaustive manner. Mean shift algorithm is a great tool which is not majorly restricted to the application of Object Tracking. but is also useful in other applications like clustering, modeling the background in video surveillance etc. As with any other algorithm, mean shift algorithm comes with many strengths and weaknesses. An Object Tracking using Mean Shift algorithm in C++ can allow real time processing of video stream 30 frames per second which seems to be computationally not so expensive . But still the classic mean shift algorithm is quite a time intensive with complexity given by $O(Tn^2)$ where T is the number of iterations and n is the number of data points in the data set. Many techniques of adaptive window(h) have been made to reduce its complexity and make it converge faster.

Further , this object tracking using Mean shift algorithm is an example of an in-situ optimization which is very powerful paradigm wherein the input domain is used to define an optimization problem and the solution is produced out of that. Mean shift algorithm is a non-parametric density gradient estimation algorithm which works on the real data without requiring any parameters and does not assume any prior shape of data clusters and can handle arbitrary feature spaces which makes it a powerful technique.

Further , the window size(h radius like ROI/window/filter mentioned in our analysis) in a mean shift algorithm needs to be carefully chosen like if we choose the larger window it can cause the merged modes (various modes together) etc. So, the point to remember is that window size needs to be carefully chosen and even we can use the adaptive window size. For example, in case of hands movement example as the tracking object (hand) size changes in each frame , so the scale of the kernel (h) needs to be adapted. There are various strategies proposed like Comaniciu's variable bandwidth methods , Rasmussen and Hager method of adding a border of pixels around the window , but we have to choose the method according to our application. Further, the Mean shift might not work well in higher dimensions as in higher dimensions , the number of local maxima is pretty high, and it might converge to a local optima soon.

Finally, from the details in this paper , it can be concluded that mean shift algorithm with its pros and cons is a valuable , versatile computational model/tool which proves to be quite useful in various computer vision applications and other areas.

6. References:

a. Mean Shift “ A Robust Approach towards Feature space Analysis “ Dorin Comaniciu , Member IEEE and Peter Meer , Senior Member , IEEE
b. Mean Shift Analysis and Applications , Dorin Comaniciu and Peter Meer Department of Electrical and Computer Engineering Rutgers University, Piscataway, NJ
c. Kernel-Based Object Tracking , Dorin Comaniciu , Visvanathan Ramesh, Peter Meer ,Real-Time Vision and Modeling Department Siemens Corporate Research 755 College Road East, Princeton, NJ 08540 and Electrical and Computer Engineering Department Rutgers University 94 Brett Road, Piscataway, NJ 08854-8058
d. “MEAN SHIFT ALGORITHM AND ITS APPLICATION IN TRACKING OF OBJECTS “ ,ZHI-QIANG WEN, ZI-XING CAI College of Information Science and Engineering, Central South University, Hu Nan, Changsha, 410083 China