SSD Tracking Using Dynamic Template and Log-polar Transformation

Qiang Zhu1 Kwang-Ting Cheng1 Hongjiang Zhang2

gzhu@ece.ucsb.edu timcheng@ece.ucsb.edu hjzhang@microsoft.com

1Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106
2Microsoft Research Asia, Beijing, P.R.China, 100080

Abstract

With the assumption of small motion displacements in sequences, templates can be matched successfully by SSD (sum-squared-difference) optimization technique. In this paper, two crucial modifications are proposed to improve the original SSD tracking. First, templates are dynamically updated for each frame. Thus, instead of designing complicated parametric models to explain various image distortions in tracking sequences, more efficient and compact parametric models, say the pure translation model, can be adopted. Second, rotation and scale can be incorporated into a translation motion model after a Log-polar transformation. Extensive experimental results performed on live video sequences show the effectiveness and advantage of our approach.

Keywords: SSD Optimization, Dynamic Template, Log-polar Transformation, Face Tracking

1. Introduction

Visual tracking has emerged as an active research field in several applications including HCI [2] [11] and video surveillance [5]. The techniques [7] [8] related to image correlation imply a good solution to this challenging problem. With the assumption of small displacements between two consecutive frames, object can be tracked successfully by optimizing some matching criterions with respect to image deformations. Hager [3] proposed a general and efficient SSD (sum-squared-difference) framework to track objects undergoing geometric distortion, changing illumination and partial occlusion. Their work further showed that the computations needed to perform SSD optimization can be factored to greatly improve algorithm efficiency. After introducing twist motion and exponential maps, Bregler [1] reformulated SSD framework to the problem of articulated human body tracking in 3D space. Most of the previous attempts mainly focused on developing more accurate and complex 2D/3D parametric models to account for the image changes caused by geometry deformations, perspective projection, non-rigid object distortions and illumination changes etc. Unfortunately, complicated parametric models may bring several unexpected results. First, considerable increases in the computational effort required to establish correspondence are conflicted with the real-time applications. Secondly, for some applications, say face tracking, pose relative to viewing camera leads to dramatic appearance changes usually failed to be modeled accurately. On the other hand, we noticed that a legal or mild human motion implies continuous image changes in sequences. Hence, instead of using complicated parametric models to interpret the image warps between the current frame and a static template obtained at the initial time, we propose a simple and low-order parametric (pure translation) model to describe the appearance changes between the current frame and a dynamic template, which is updated according to the content in the previous frame. Furthermore, by a Log-polar transformation, a pure translation model can account for the image variations caused by rotation and scale.

The remainder of this paper is organized as follows. In section 2, we show the principle of SSD optimization. Then, the comparisons between two types of SSD trackers (static template plus affine motion model vs. dynamic template plus pure translation model) will be performed in detail. Section 3 introduces Log-polar transformation to incorporate the rotation and scale into the tracker with the pure translation model. Section 4 shows the work of a real-time tracking platform. Finally, we present a short discussion of conclusion.

2. SSD Region Tracking

2.1 Algorithm definition

The 2D changes of image appearance under orthographic projection can be viewed as affine motion described by a six-parameter vector \( U(a1, a2, a3, a4, dx, dy) \). For a point \((x, y)\) at time \(t\), the new point \((x_{t+1}, y_{t+1})\) at time \(t+1\) after an affine motion can be defined as:

\[
\begin{bmatrix}
x_{t+1} \\
y_{t+1}
\end{bmatrix} = A \times \begin{bmatrix}
x_t \\
y_t
\end{bmatrix} + \begin{bmatrix}
dx \\
dy
\end{bmatrix} = \begin{bmatrix}
a_1 & a_2 & dx \\
a_3 & a_4 & dy
\end{bmatrix} \times \begin{bmatrix}
x_t \\
y_t
\end{bmatrix} + \begin{bmatrix}
dx \\
dy
\end{bmatrix}
\]

Then the SSD objective function can be evaluated with respect to a set of points in the target region \( R \):

\[
O(u) = \sum_{x \in R} (I(f(P, U), t+1) - I(P, t))^2
\]

where \( I(P, t) \) is the image intensity value of a point \( P \) of \( f(P, t) \) indicates the affine transformation for \( P \) from time \( t \) to \( t+1 \). The linearization of Equation (2) can be written by expanding \( I(f(P, U), t+1) \) in a Taylor series with respect to motion \( U \) and time \( t \):

\[
O(u) = \sum_{x \in R} (L_{ux} + L_{ux} + L_{ux} + L_{ux} + L_{dx} + L_{dx} + L_{dy} + L_{dy})^2
\]

Optimization above function yields the solution:

\[
\Delta U = -(M^T M)^{-1} M^T I
\]
M is the Jacobian matrix of image partial derivative (Ia1, Ia2, Ia3, Ia4, Idx, Idy) at time t and plays a central role in SSD tracking. In general, the image after inverse warping should satisfy the continuous time-varying character formulated below:

\[ I = MU + I_0 \]  

This image constancy assumption is equal to a parameterized version of Horn’s optical flow constraint [4]. In this description, Jacobian matrix M is to relate the image changes caused by the spatial motion to the temporal variations. In the case of pure translation, the motion Equation (1) is simplified and only (dx, dy) is remained while computing the new point position.

2.2 Motion template

Each component in the Jacobian matrix M can be treated as one unique motion template. Thus, Equation (5) can be described by a weighted combination of all motion templates to explain the temporal variation of image intensity, i.e. I. In this sense, the whole SSD optimization can be treated as an iterative process of searching the best weight set, i.e. the values of incremental motions. For different motion templates, we can divide them into two categories. For pure translations along vertical and horizontal directions, they are equal to the basic image gradients. Hence, the variations caused by these two motion templates are totally coincident with the image change itself. For other complicated motions, say rotation and scale, the motion templates are defined as a combination of the image gradients and the partial derivative terms. This difference between the two categories clarifies the fact that motion templates in first category are more intuitive and robust cues to interpret the image changes.

2.3 Implementation and evaluation

In Fig. 1 and Fig. 2, we implemented two types of SSD tracker based on different strategies. STAM, Static Template & Affine Motion tracker, is using an affine motion model to explain the image changes between the current frame and a static template initialized from the first frame. DTTM, Dynamic Template & Translation Model tracker, is using a pure translation motion model to interpret the image variations between the current frame and a dynamic template updated for each frame.

For each frame in image sequences:
1) evaluating I between two consecutive frames
2) with equation (4), \( \Delta U \) calculated
3) if \( \Delta U > \text{threshold} \), GOTO Step 1
4) updating dynamic template M according to the motion templates (only Idx and Idy are considered) with respect to the current frame

**Figure 2: DTTM tracker**

a) Face video with dramatic pose variations, we showed the results of 40f, 52f, 78f, 108f, 126f, 197f, 234f, 331f, 366f.

b) Face video with illumination changes and translation, we showed the results of 1f, 40f, 44f, 50f, 65f, 70f, 80f, 92f, 98f.

(For STAM tracker, target will be lost in the 67th and 12th frame in these two videos. Due to the page limitation, we don’t present the visual results for this tracker.)

**Figure 3: Some tracking results of DTTM tracker**

We tested the two trackers by two typical video clips under various situations like dramatic pose variations, fast translation and illumination changes. In our implementation, we choose the region containing the mouth and the nose as our target, since this region
contains abundant and stable image gradients. In our experiments, we observe that the DTTM tracker works much more robust than the STAM tracker. Moreover, the computation efficiency is also an important factor to be considered, which implies a small region preferred for time-sensitive applications. The DTTM tracker is twice as fast as the STAM due to the extra affine warp computations required for the latter.

Now, dynamical template technique is verified to be superior to that of static template, we further observe the different performance between an affine motion model and a pure translation model if dynamical template is adopted. Basically, the SSD residual value, i.e. the $O(u)$ defined in Equation (2), intuitively reflects the ability for a motion model to interpret the image changes after a homogenous transformation between two consecutive frames. In Fig. 4, the red curve depicts the SSD residuals if a pure translation model is adopted, while the blue curve indicates the SSD residuals of an affine motion model. We can observe the two curves are pretty close, which implies these two motion models have the similar tracking accuracy. On the other hand, the affine motion model obviously requires more computation time and leads to hard convergence. Hence, we believe the dynamic template technique plus the affine motion model is the best choice for SSD tracking, in the senses of both the tracking accuracy and the computation efficiency.

For each frame in image sequences:
I. Applying DTTM tracker to the original image.
II. Transforming image from Cartesian space to Log-polar space.
III. Applying DTTM tracker to the new generated image in Log-polar space.
IV. Transforming incremental motions from Log-polar coordinates to Cartesian coordinates.

Figure 5: DTTM tracker

4. Real-time Attention Tracking System

Based on the developed tracker, we built a real-time ATS (Attention Tracking System) platform by positioning a web-camera atop the computer monitor. In our system, a powerful and efficient nose detector [9] is implemented to initialize the system, and reset the system while full occlusions or exceptional SSD residuals occurred. As a core part of the ATS platform, DTTM and EDTTM tracker are used to track the human face in the video sequences. We also developed a pose estimation algorithm [12], and hence the ATS platform is able to report the attention direction of a computer user in real-time with single camera. The attentive document window is activated automatically when the user turns to it and the mouse/keyboard cursor is put at the original position automatically, so that the user is able to switch among multiple windows without mouse motion. The system achieves real-time (15frame/s) performance on a P4 2GHz CPU and a typical USB web camera with resolution of 320*240 pixel is used.

Figure 6 presents a full test on our ATS platform including various types of face motions like translation, yaw and in-plane rotation. In addition, dramatic face expressions, partial occlusions are contained in this video. As it shown, the whole system works pretty well and the tracking results seem quite stable in a long image sequence. Our powerful nose detector can resume the whole system in time of need. Finally, the head pose, indicated by two bars on the left and bottom, is showed in the screen simultaneously.
1th row: initialization and big yaw
2th row: big tilt and in-plane rotation
3th row: robust tracking while opening mouth
4th row: disturbed by tongue
5th row: serious occlusions caused by human hand

Figure 6: Full test of our tracker on AFT platform

5. Conclusions and Future Works

In this paper, we proposed robust and efficient SSD trackers using dynamic template technique and Log-polar transformation. According to the extensive experimental results, we validated that dynamic template technique plus a compact translation model is superior to other combinations, say static template plus an affine motion model. Further, we applied Polar-log transformation to the original image, and then we can report more types of motions (rotation and scale) without introducing more complicated motion models. In section 4, we show the effectiveness and advantage of our tracker by integrating it into the Attention Tracking System (ATS) platform.

For future works, we will consider incorporating our region tracking algorithm into a global 3D probabilistic framework (particle filter [6], for example) to overcome some of the difficulties inherent in simple 2D tracking techniques. Moreover, the methods to combine multi-clues (shape, color and appearance) can be further studied as a co-inference tracking problem [10].

Reference


