Object Tracking in Videos Supplementary Information

Supplementary Info

Here we introduce the details of the Lucas-Kanade Tracker and the Matthews-Baker Tracker algorithms. You may refer to this material for the assignment.

Lucas-Kanade Tracker

A Lucas Kanade tracker uses a warp function $\mathbf{W}(\mathbf{x};\mathbf{p})$ to align an image $\mathbf{I}$ to a template $\mathbf{T}$. The optimal parameters $\mathbf{p}^*$ of the warp $\mathbf{W}(\mathbf{x};\mathbf{p})$ are found by minimizing loss $\mathcal{L}$. The loss $\mathcal{L}$ is the pixel-wise sum of square difference between the warped image $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ and template $\mathbf{T}$, refer eq 1 and 2.

Note: We denote pixel locations by $\mathbf{x}$, so $\mathbf{I}(\mathbf{x})$ is the pixel value (eg. *rgb*) at location $\mathbf{x}$ in image $\mathbf{I}$. For simplicity, $\mathbf{I}$ and $\mathbf{T}$ are treated as column vectors instead of a matrix (like linearized matrices!). $\mathbf{W}(\mathbf{x};\mathbf{p})$ is the point obtained by warping $\mathbf{x}$. $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ is the pixel value (eg. *rgb*) after warping.

$$\mathcal{L} = \sum_{\mathbf{x}} \left[\mathbf{T}(\mathbf{x}) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))\right]^2 \tag{1}$$

$$\mathbf{p}^* = \underset{\mathbf{p}}{\operatorname{argmin}}\, \mathcal{L} \tag{2}$$

In general, this is a difficult non-linear optimization of $\mathcal{L}$ in $\mathbf{p}$ for even simple warps like *affine warps*, but the Lucas-Kanade tracker circumvents this by using a key idea - **forward additive alignment**. The idea assumes we already have a very close estimate $\mathbf{p}_0$ of the correct warp $\mathbf{p}^*$, then we can assume that a small linear change $\Delta\mathbf{p}$ is enough to get the best alignment i.e. $\mathbf{p}^* = \mathbf{p}_0 + \Delta\mathbf{p}$. This is the forward additive form of the warp. The loss $\mathcal{L}$ can then be written as:

$$\mathcal{L} = \sum_{\mathbf{x}} \left[\mathbf{T}(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p}))\right]^2 \tag{3}$$

$$\Delta\mathbf{p}^* = \underset{\Delta\mathbf{p}}{\operatorname{argmin}}\, \mathcal{L} \tag{4}$$

$$\mathbf{p}^* = \mathbf{p}_0 + \Delta\mathbf{p}^* \tag{5}$$

Expanding this to the first order with Taylor Series:

$$\mathcal{L} \approx \sum_{\mathbf{x}} \left[\mathbf{T}(\mathbf{x}) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}_0)) - \nabla\mathbf{I}(\mathbf{x})\frac{\partial\mathbf{W}}{\partial\mathbf{p}_0}\Delta\mathbf{p}\right]^2 \tag{6}$$

Here the Jacobian of $\mathbf{I}(\mathbf{x})$, $\nabla\mathbf{I}(\mathbf{x}) = \left[\frac{\partial\mathbf{I}(\mathbf{x})}{\partial u}\, \frac{\partial\mathbf{I}(\mathbf{x})}{\partial v}\right]$, is the vector containing the horizontal and vertical gradient at pixel location $\mathbf{x}$. Rearranging the Taylor expansion, it can be rewritten as a typical least squares approximation $\Delta\mathbf{p}^* = \underset{\Delta\mathbf{p}}{\operatorname{argmin}} ||A\Delta\mathbf{p} - b||^2$, note we pull out a negative sign thanks to the squaring,

$$\Delta\mathbf{p}^* = \underset{\Delta\mathbf{p}}{\operatorname{argmin}} \sum_{\mathbf{x}} \left[\nabla\mathbf{I} \frac{\partial\mathbf{W}}{\partial\mathbf{p}_0} \Delta\mathbf{p} - \{\mathbf{T}(\mathbf{x}) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}_0))\} \right]^2 \tag{7}$$

This can be solved with $\Delta\mathbf{p}^* = (A^T A)^{-1} A^T b$ where $A^T A$ is the Hessian $\mathbf{H}$:

$$(A^T A) = \mathbf{H} = \sum_{\mathbf{x}} \left[\nabla\mathbf{I} \frac{\partial\mathbf{W}}{\partial\mathbf{p}_0} \right]^T \left[\nabla\mathbf{I} \frac{\partial\mathbf{W}}{\partial\mathbf{p}_0} \right] \tag{8}$$

$$A = \left[\nabla\mathbf{I} \frac{\partial\mathbf{W}}{\partial\mathbf{p}_0} \right] \tag{9}$$

$$b = \mathbf{T}(\mathbf{x}) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}_0)) \tag{10}$$

Once $\Delta\mathbf{p}^*$ is computed, the best estimate warp $\mathbf{p}^*$ can be estimated using eq 5. However, in practice, our initial estimate $\mathbf{p}_0$ can be well off the optimal $\mathbf{p}^*$, thus violating the **forward additive alignment** assumption. As a fix, we perform the optimization $\mathcal{L}$ in an iterative fashion using an error threshold ϵ as described in algorithm 2 from [**?**].

Algorithm 1 The Lucas-Kanade Algorithm

---

(0) $\mathbf{p} \leftarrow \mathbf{p}_0$

(1) Iterate until $||\Delta\mathbf{p}|| \leq \epsilon$:

(2) Warp $\mathbf{I}$ with $\mathbf{W}(\mathbf{x};\mathbf{p})$ to compute $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$

(3) Compute the error image $\mathbf{E}(\mathbf{x}) = \mathbf{T}(\mathbf{x}) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$

(4) Warp the gradient $\nabla\mathbf{I}$ with $\mathbf{W}(\mathbf{x};\mathbf{p})$

(5) Evaluate the Jacobian $\frac{\partial\mathbf{W}}{\partial\mathbf{p}}$ at $(\mathbf{x};\mathbf{p})$

(6) Compute the steepest descent image $\nabla\mathbf{I}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}$

(7) Compute the Hessian matrix $\mathbf{H} = \sum_{\mathbf{x}} \left[\nabla\mathbf{I}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T \left[\nabla\mathbf{I}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]$

(8) Compute $\Delta\mathbf{p} = \mathbf{H}^{-1} \sum_{\mathbf{x}} \left[\nabla\mathbf{I}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T \mathbf{E}(\mathbf{x})$

(9) Update the parameters $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$

Matthews-Baker Tracker

While the Lucas-Kanade tracker works very well, it is computationally expensive due to the computation of the Hessian and Jacobian for each frame of the video. The Matthews-Baker tracker is similar to the Lucas-Kanade tracker, but requires less computation, as the Hessian and Jacobian are only computed once for the video. The Matthews-Baker tracker replaces **the forward additive alignment** idea of the Lucas-Kanade tracker by the **inverse compositional alignment**. It assumes that the warp $\mathbf{W}(\mathbf{x};\mathbf{p})$ is an invertible function. Since *the affine warps* used by the Lucas-Kanade tracker are indeed invertible (why?, think!), we can use the Matthews-Baker tracker instead of a Lucas-Kanade tracker.

The key to the efficiency is switching the role of the image $\mathbf{I}$ and the template $\mathbf{T}$ in the algorithm. In contrast to the Lucas-Kanade tracker (warp image $\mathbf{I}$ to template T), the Matthews-Baker tracker warps the template $\mathbf{T}$ to the image $\mathbf{I}$. This key difference leads to the computational gains because unlike the image $\mathbf{I}$ which changes with each frame of the video, the template $\mathbf{T}$ remains fixed throughout the video. If we can write the Hessian and Jacobian involved in the optimization of $\mathcal{L}$ in terms of the template T

instead of image I, then we only need to compute them once at the start of the tracking.

Here is the inverse compositional form of the Matthew-Baker tracker given without the proof of equivalence to the forward additive form of the Lucas-Kanade tracker. Please refer [?] for the proof. Given an initial estimate $\mathbf{p}_0$, we want to find the $\Delta\mathbf{p}^*$ to minimize $\mathcal{L}$ as follows,

$$\mathcal{L} = \sum_{\mathbf{x}} \left[\mathbf{T}\left(\mathbf{W}\left(\mathbf{x}; \Delta\mathbf{p}\right)\right) - \mathbf{I}\left(\mathbf{W}\left(\mathbf{x}; \mathbf{p}_0\right)\right)\right]^2 \tag{11}$$

$$\Delta\mathbf{p}^* = \operatorname*{argmin}_{\Delta\mathbf{p}} \mathcal{L} \tag{12}$$

This completes our role switch of the template $\mathbf{T}$ and the image $\mathbf{I}$, please note that $\Delta\mathbf{p}$ are the parameters of the warp $\mathbf{W}$ applied to $\mathbf{T}$ and $\mathbf{p}_0$ are the parameters of the warp $\mathbf{W}$ applied to $\mathbf{I}$.

Another key difference of Matthews-Baker tracker to the Lucas-Kanade tracker is - how do we combine the two warps $\mathbf{W}(\mathbf{x};\mathbf{p}_0)$ and $\mathbf{W}(\mathbf{x};\Delta\mathbf{p})$? In the Lucas-Kanade tracker we combined the two warps by simply adding parameter $\mathbf{p}_0$ to another parameter $\Delta\mathbf{p}$, and produce a new warp $\mathbf{W}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p})$. Specifically, $\mathbf{p}^* = \mathbf{p}_0 + \Delta\mathbf{p}^*$, $\mathbf{W}(\mathbf{x};\mathbf{p}^*) = \mathbf{W}(\mathbf{x};\mathbf{p}_0) + \mathbf{W}(\mathbf{x};\Delta\mathbf{p}^*)$. In Matthews-Baker tracker, we use another way of combination through composition as follows,

$$\mathbf{W}(\mathbf{x};\mathbf{p}^*) = \mathbf{W}(\mathbf{x};\mathbf{p}_0) \circ \mathbf{W}(\mathbf{x};\Delta\mathbf{p}^*)^{-1} \tag{13}$$

If we are using *affine warps*, then the warps can be implemented as matrix multiplication, then we can simplify eq 13 as follows

$$\begin{aligned}\mathbf{W}(\mathbf{x};\mathbf{p}^*) &= \mathbf{W}(\mathbf{W}(\mathbf{x};\Delta\mathbf{p}^*)^{-1};\mathbf{p}_0)\\ \mathbf{W}(\mathbf{x};\mathbf{p}^*) &= \mathbf{W}(\mathbf{p}_0)\mathbf{W}(\Delta\mathbf{p}^*)^{-1}\mathbf{x}\end{aligned}$$

Let us simplify $\mathcal{L}$ as before using first order linear approximation of $\mathbf{T}\left(\mathbf{W}\left(\mathbf{x};\Delta\mathbf{p}\right)\right)$. Please note, for tracking a template $\mathbf{T}$, the summation over $\mathbf{x}$ is performed only over the pixels lying inside $\mathbf{T}$.

$$\mathcal{L} \approx \sum_{\mathbf{x}} \left[\mathbf{T}\left(\mathbf{x}\right) + \nabla\mathbf{T}(\mathbf{x})\frac{\partial\mathbf{W}}{\partial\mathbf{p}_0}\Delta\mathbf{p} - \mathbf{I}\left(\mathbf{W}\left(\mathbf{x};\mathbf{p}_0\right)\right)\right]^2 \tag{14}$$

where $\nabla\mathbf{T}(\mathbf{x}) = \left[\frac{\partial\mathbf{T}(\mathbf{x})}{\partial u}\ \frac{\partial\mathbf{T}(\mathbf{x})}{\partial v}\right]$.

We now proceed to find a closed form solution to $\Delta\mathbf{p}^*$ by equating the derivative $\frac{\partial \mathcal{L}}{\partial \Delta \mathbf{p}}$ to zero.

$$\frac{\partial \mathcal{L}}{\partial \Delta \mathbf{p}} = 2\sum_{\mathbf{x}} \left[\nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}_0}\right]^T \left[\mathbf{T}(\mathbf{x}) + \nabla \mathbf{T}(\mathbf{x}) \frac{\partial \mathbf{W}}{\partial \mathbf{p}_0} \Delta \mathbf{p} - \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}_0))\right] \tag{15}$$

Setting to zero, switching from summation to vector notation and solving for $\Delta\mathbf{p}$ we get

$$\Delta \mathbf{p} = \mathbf{H}^{-1} \mathbf{J}^T \left[\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}_0)) - \mathbf{T}\right] \tag{16}$$

where $\mathbf{J}$ is the Jacobian of $\mathbf{T}(\mathbf{W}(\mathbf{x}; \Delta\mathbf{p}))$, $\mathbf{J} = \nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}_0}$, $\mathbf{H}$ is the approximated Hessian $\mathbf{H} = \mathbf{J}^T \mathbf{J}$ and $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}_0))$ is the warped image. Note that for a given template, the Jacobian $\mathbf{J}$ and Hessian $\mathbf{H}$ are independent of $\mathbf{p}_0$. This means they only need to be computed once and then they can be reused during the entire tracking sequence.

Once $\Delta\mathbf{p}^*$ has been solved for, it needs to be inverted and composed with $\mathbf{p}_0$ to get the new warp parameters for the next iteration.

$$\mathbf{W}(\mathbf{x}; \mathbf{p}_0) \leftarrow \mathbf{W}(\mathbf{x}; \mathbf{p}_0) \circ \mathbf{W}(\mathbf{x}; \Delta\mathbf{p}^*)^{-1} \tag{17}$$

The next iteration solves Equation 16 starting with the new value of $\mathbf{p}_0$. Possible termination criteria include the absolute value of $\Delta\mathbf{p}^*$ falling below some value or running for some fixed number of iterations.

Algorithm 2 The Matthews-Baker Algorithm

(0) $\mathbf{p} \leftarrow \mathbf{p}_0$

(1) Pre-compute

(1) Evaluate the gradient of $\nabla \mathbf{T}$ of the template $\mathbf{T}(\mathbf{x})$

(2) Evaluate the Jacobian $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ at $(\mathbf{x}; \mathbf{0})$

(3) Compute the steepest descent images $\nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$

(4) Compute the Hessian matrix using $\mathbf{J} = \nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}, \mathbf{H} = \mathbf{J}^T \mathbf{J}$

(5)

(6) Iterate until $||\Delta\mathbf{p}|| \leq \epsilon$:

(7) Warp $\mathbf{I}$ with $\mathbf{W}(\mathbf{x}; \mathbf{p})$ to compute $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$

(8) Compute the error image $\mathbf{E}(\mathbf{x}) = \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) - \mathbf{T}(\mathbf{x})$

(9) Compute $\Delta \mathbf{p} = \mathbf{H}^{-1} \sum_{\mathbf{x}} \left[\nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right]^T \mathbf{E}(\mathbf{x})$

(10) Update the parameters $\mathbf{p} \leftarrow \mathbf{p} \circ (\Delta \mathbf{p})^{-1}$