Continuous Event-Line Constraint for Closed-Form Velocity Initialization

BMVC 2021 Submission # 877

1 Details of Our M-estimator Implementation

As mentioned in the paper, the details of our implementation can be found in the *Definition* part of *Chapter 1.3* in [1]. We also provide some details here.

For the classical regression model, the response variable Y is a linear combination of the predictor variables β, which is given by the matrix form

$$Y = X\beta + e, \tag{1}$$

where e is a random errors' vector, which is usually assumed to be independent and normally distributed with expected value 0 and unknown dispersion σ. In our case, Y is a vector with all elements 0.

A least squares method minimizes the residual sum of squares $Q = \sum_{i=1}^{n}(Y_i - \sum_{j=1}^{m} x_{ij}\beta_j)^2 = \sum_{i=1}^{n} e_i^2$. It can be concluded from this formula that outliers have a great influence on system estimation. The basic idea of M-estimated robust regression is to use iterative weighted least squares to estimate the regression coefficient, and determine the weight of each sample according to the regression residual to achieve the purpose of robustness. The optimization objective function is

$$Q = \sum_{i=1}^{n} \rho(e_i) = \sum_{i=1}^{n} \rho(Y_i - \sum_{j=1}^{m} x_{ij}\beta_j), \tag{2}$$

here, ρ called *influence function* (IF), and we adopt the one proposed by Huber [2]

$$\rho(x) = \begin{cases} x^2/2, & |x| \leq k \\ k|x| - k^2/2, & |x| > k \end{cases}, \tag{3}$$

where we take the widely used 1.345 for k.

Taking the partial derivative of Q with respect to β and setting the partial derivative equal to zero, we obtain

$$\sum_{i=1}^{n} \psi(Y_i - X_i\beta)X_i = 0, \tag{4}$$

where $\psi(x)$ is the derivative of $\rho(x)$.

In order to make the M-estimator robust, a robust scale estimate s is introduced in the weight function to standardize the residuals, namely e_i/s. s is generally taken as the median absolute deviation (MAD) divided by a constant 0.6745, which is proposed by Hampel [1]. We then get the standardized residuals $u_i = e_i/s = 0.6745e_i/\text{med}(|e_i|)$, where med stands for median here.

By equation (4), we then get

$$\sum_{i=1}^{n} \psi(u_i)X_i = \sum_{i=1}^{n} \frac{\psi(u_i)}{u_i} u_i X_i = W_i u_i X_i = 0, \tag{5}$$

W_i here is the weight of the sample with index i.

After vectorization, by equation (1), we obtain

$$X^{\mathrm{T}}WY = X^{\mathrm{T}}WX\beta. \tag{6}$$

Therefore, the specific algorithm steps are as follows

1. Select the result by calculating the SVD of $X^{\mathrm{T}}X$ (as mentioned in our paper) as the initial iteration value $\hat{\beta}^{(0)}$, and obtaining the initial residual e.
2. After standardized residuals, u is obtained. We then obtain the initial weight of each sample by $W_i = \frac{\psi(u_i)}{u_i}$.
3. Next, we get $\hat{\beta}^{(1)}$ by calculating the SVD of $X^{\mathrm{T}}WX$ instead of $\hat{\beta}^{(0)}$ to get the new residual e, thereby obtaining new weights.
4. Return to step 2, and calculate the robust estimation value $\hat{\beta}^{(i)}$ by analogy. When the maximum absolute value of the difference between the regression coefficients of two adjacent steps is less than the set standard error, end the iteration (i.e. $\max(|\hat{\beta}^{(i)} - \hat{\beta}^{(i-1)}|) < \varepsilon$).

2 Degenerated Case of The Linear Solver

Note that the linear solver can not always uniquely determine the speed. It is obvious that if the motion of the camera is a pure rotation, which means $\mathbf{v} = \mathbf{0}$, solving our linear equation (9) via SVD will not work.

Besides, if the camera moves along a straight line without any rotation, there exists a further degenerate case. This is obvious because any translational velocity component along the direction of the 3D line $L = \mathbf{l}_1 \times \mathbf{l}_3$ will not contribute to any appearance changes in the image, and therefore also no events. Hence, the 3D line direction needs to lie in the null space of the matrix $\mathbf{A}$, which means $\mathbf{A}(\mathbf{l}_1 \times \mathbf{l}_3) = 0$. Moreover, $\mathbf{A}\mathbf{v} = \mathbf{A}(\mathbf{v}_1 + \mathbf{l}_1 \times \mathbf{l}_3) = \mathbf{A}\mathbf{v}_1$. There exists an unobservable direction for the translational velocity given by the direction of the 3D line. The unobservable direction also exists with linear motion when there are multiple lines but all of them are parallel. Degenerate cases are well understood if considering the degenerate cases of the classical tri-focal tensor geometry.

References

[1] Frank R Hampel. The breakdown points of the mean combined with some rejection rules. *Technometrics*, 27(2):95–107, 1985.

[2] Peter J Huber. *Robust statistics*, volume 523. John Wiley & Sons, 2004.

[3] Andreas Ruckstuhl. Robust fitting of parametric models based on m-estimation. *Lecture notes*, page 40, 2014.