Focus of Expansion Localization through Inverse C-Velocity

Adrien Bak¹, Samia Bouchafa¹, and Didier Aubert²

¹ Institut d'Electronique Fondamentale, Université Paris-Sud, Bat 220
91405, Orsay, France

{adrien.bak,samia.bouchafa}@ief.u-psud.fr

² IFSTTAR

58 Bld Lefebvre, 75015 Paris, France

aubert@ifsttar.fr

Abstract. The Focus of Expansion (FoE) sums up all the available information on translational ego-motion for monocular systems. It has also been shown to present interesting features in cognitive research. As such, its localization bears great importance, either for robotic applications, as well as for attention fixation research. It will be shown that the so-called C-Velocity framework can be inverted in order to extract the FoE position from a rough scene structure estimation. This method rely on robust cumulative framework and only exploit the optical flow field relative norm as such, it is robust to angular noise and bias on the absolute optical flow norm.

1 Introduction – State of the Art

The Focus of Expansion holds a particular importance for monocular vision systems. Its position contains all the available information on the translational

ego-motion $\mathbf{T} = \begin{vmatrix} T_X \\ T_Y \\ T_Z \end{vmatrix}$:

$$\mathbf{FoE} = \begin{vmatrix} x_{FoE} = f \frac{T_X}{T_Z} \\ y_{FoE} = f \frac{T_Y}{T_Z} \end{vmatrix} \quad (1)$$

Knowing its position is useful for a wide variety of problems, such as path planification or collision avoidance [2][3]. The importance of the FoE is not limited to computer vision or robotics. Indeed, it has been shown that the FoE holds a strong importance in the study of behaviour, cognition, and more specifically, visual attention [4].

As such, the localization of the FoE has always been an active research topic. Early methods [5][6] relied on the extraction and matching of image primitives. Such a discrete approach was dictated by the computational costs and reliability of early optical flow implementation. As such, those were not very robust due to the fact that they used only partial information.

With the emergence of efficient optical flow algorithms, global methods were introduced. Most methods rely on the fact that the Focus of Expansion is point

of convergence of all optical flow field lines. For example, the authors of [7] use a voting scheme in order to locate it. In a similar manner, the work presented in [8] presents a matched filter that exploits the change of sign of the optical flow component around the FoE to detect it. An interesting approach, based on vector calculus, was unveiled in [9]. However, due to the fact that it exploits the computation of divergence and curl of the optical flow, it is highly sensitive to noise. Finally, methods based purely on geometric concerns used to be prohibitive, due to their computational costs, authors of [10] present a novel way to take geometric constraints into account, but the computation cost is still unfitted to real-time applications.

All these methods make use of both the angular and the radial components of the optical flow, at least implicitly. However, as it will be shown in this study, this is not necessary. Radial component only is sufficient to extract the position of the FoE. Actually, the radial component is only needed up to a scale factor. This makes the proposed method insensitive to noise on the angular component and to bias on the radial component. The former being the prominent one in most real-time capable algorithms, especially on image discontinuities [11], the later being a common pitfall of local optical flow estimation methods.

Contrary to previous works, the method presented here stems from a scene structure estimation method. As such, the method proposed here is the second step toward an integrated monocular system allowing to determine both scene structure and translational ego-motion without relying on feature extraction or additional hardware.

In the following section, the basic hypothesis will be detailed, the initial Direct C-Velocity framework will be shortly described. Then the proposed method, Inverse C-Velocity, will be unveiled. Results obtained on simulated and real data will be presented. Finally, these results will be discussed and some future developments will be presented.

2 C-Velocity Reminder

2.1 Basis – Notations

A vision sensor, attached to a mobile (*e.g.* a car) is moving in a world consisting of dynamic and static objects.

The sensor is represented by the pinhole model, its focal length is denoted f .

Between $t = t_0$ and $t = t_1$ the sensor undergoes an arbitrary translational

motion $\vec{T} = \begin{vmatrix} T_X \\ T_Y \\ T_Z \end{vmatrix}$. A static world point M is located (in a sensor-centered frame

of reference) in $\begin{vmatrix} X_M \\ Y_M \\ Z_M \end{vmatrix}$ at $t = t_0$. At $t = t_1$, its coordinates are $\begin{vmatrix} X_M - T_X \\ Y_M - T_Y \\ Z_M - T_Z \end{vmatrix}$. Its

image at $t = t_0$ being $m = \begin{vmatrix} x \\ y \end{vmatrix}$, the optical flow in m can be expressed as:

$$\begin{aligned}\mu &= \frac{\partial}{\partial t} \left(f \frac{X_M}{Z_M} \right) \approx f \left(\frac{\Delta X_M}{Z_M} - X_M \frac{\Delta Z_M}{Z_M^2} \right) \\ \nu &= \frac{\partial}{\partial t} \left(f \frac{Y_M}{Z_M} \right) \approx f \left(\frac{\Delta Y_M}{Z_M} - Y_M \frac{\Delta Z_M}{Z_M^2} \right)\end{aligned}\quad (2)$$

Where $\Delta A = A(t_1) - A(T_0)$.

Under classical assumptions, the optical flow, at a point $m = \begin{vmatrix} x \\ y \end{vmatrix}$ can then be written:

$$\begin{cases} \mu = \frac{xT_Z - fT_X}{Z} \\ \nu = \frac{yT_Z - fT_Y}{Z} \end{cases}\quad (3)$$

Let us now consider a planar surface of equation:

$$\mathbf{n}\mathbf{P} = d \quad (4)$$

with $\mathbf{n} = \begin{vmatrix} n_x \\ n_y \\ n_z \end{vmatrix}$ its normal vector, $\mathbf{P}$ the position vector of a particular world point and d the distance between the plane and the origin. The motion field, for this particular plane is given by, when combining 1 and 2:

$$\begin{cases} \mu = \frac{1}{fd} (a_1x^2 + a_2xy + a_3fx + a_4fy + a_5f^2) \\ \nu = \frac{1}{fd} (a_1xy + a_2y^2 + a_6fy + a_7fx + a_8f^2) \end{cases}\quad (5)$$

where:

$$\begin{aligned}a_1 &= T_Z n_x & a_2 &= T_Z n_y & a_3 &= T_Z n_z - T_X n_x & a_4 &= T_X n_y \\ a_5 &= T_X n_z & a_6 &= T_Z n_z - T_Y n_y & a_7 &= T_Y n_x & a_8 &= T_Y n_z\end{aligned}$$

2.2 Direct C-Velocity

Direct C-Velocity is a method that aims at detecting particular scene planes¹, representative of an urban environment through an analysis of motion, especially, optical flow:

Plane Type	$\mathbf{n}$	Distance to Origin
Road	$[0, 1, 0]^T$	d_r
Building	$[1, 0, 0]^T$	d_b

In the following, a Building plane will be taken as an example, generalization is immediate.

From Equation 5:

$$\begin{cases} \mu = \frac{x}{fd} (x - x_{FoE}) \\ \nu = \frac{x}{fd} (y - y_{FoE}) \end{cases}\quad (6)$$

The two equations 6 can be combined by writing: $w = \sqrt{\mu^2 + \nu^2}$

$$w = \frac{T_Z \|x\|}{fd} \|\mathbf{m} - \mathbf{FoE}\| \quad (7)$$

¹ The original C-Velocity work consider a third plane type 'Obstacle', however, it is not relevant for this study.

At this point, so-called *c-values* can be defined, each being specific of a particular plane hypothesis.

Plane Hypothesis	<i>C-Value</i>	Notation
Building	$ x \ \mathbf{m} - \mathbf{FoE}\ $	$c_{building}$
Road	$ y \ \mathbf{m} - \mathbf{FoE}\ $	c_{road}

These *c-values* depend only on the coordinates of the studied point and on the coordinate of the FoE.

It is noticeable that, in the case of a point that actually belongs to a building plane:

$$w \propto c_{Building} \quad (8)$$

Direct C-Velocity uses this relation in order to establish a generalized hough transform in which, every image point would vote for a set of curves able to enforce its *w* value. A further processing step (clustering, Hough-Transform) would later on be used in order to extract dominant lines, which represent scene planes in the corresponding C-Velocity voting space. This whole method was based on prior knowledge of the position of the FoE. It is worth noting that, due to numerical constraints (*c-values* can typically be in the range of hundreds of thousands), the actual relation to be used is:

$$w \propto \kappa^2 \quad (9)$$

with $\kappa = \sqrt{c}$.

Fig. 1. Direct C-Velocity Workflow

Figure 1 illustrates this. The starting point is a computed optical flow and a known FoE. From these information, voting spaces are populated. In the presented case, only one road plane is present, in the road voting space, it will form a perfect parabola, when in the building voting space, it will generate incoherent noise. Coherent parabolas can then be extracted in order to detect and locate scene planes.

One can refer to [1] for further details.

3 Inverse C-Velocity

Inverse C-Velocity objective is the exact opposite. Instead of using ego-motion prior information to extract scene structure information, scene structure prior information will be used in order to extract translational ego-motion. First, the scene structure information will be briefly discussed, then, the method for localizing the FoE will be exposed.

3.1 Scene Structure Evaluation

This system relies on the identification of sets of coplanar points (at least one) and on the knowledge of their relative normal vectors. The actual plane equations are not needed, neither are the distance from the planes to the sensor. The basic requirement for a useful scene structure evaluation system is that it provides a set of label, their corresponding plane types and a map of these labels in the image space. Not all image points need to be labelled, and not all plane categories need to be represented.

Several systems could fulfil these requirements, just to cite a few: a calibrated LIDAR system [12], a stereo-system [13], or the original C-Velocity Framework [14]. For practical reasons, in this study, a stereo-system will be used. Both V-Disparity [15] and a Generalized Hough Transform, very close to the one presented in [13] will be used in order to detect, respectively, the road plane and building planes.

However, it is important to note that, in this study, stereo-vision was only used as a plane detector, the actual depth of individual pixels was not used. Furthermore, in future developments, a purely monocular version of this system, that uses Direct C-Velocity in order to extract scene structure and Inverse C-Velocity in order to compute the ego-motion will be investigated.

Fig. 2 presents the results of such an extraction. Results, especially on the road plane, might appear sparse. Indeed, the poorly textured nature of the road plane leads to mismatches in the disparity maps. It is important to note that

Fig. 2. Image Extracted from a real sequence. Overlays: Red - the building plane ; Green - the road plane.

Fig. 3. Image extracted from a Sivic-generated sequence. Overlays: Red - the building plane ; Green - the road plane.

it is not relevant, for this study, to strictly identify a plane. What is relevant however, is to ensure that all points sharing the same label (*e.g.* red or green in the above example) belong to the same plane and that this plane validates one of the basic hypothesis (*e.g.* road or building). This drives the setup of the plane detection system toward reliability and not exhaustiveness.

3.2 Focus of Expansion Extraction

The key idea of Inverse C-Velocity is to find a metric that reflects the quality of a given FoE candidate, based on the representation exposed in the Direct C-Velocity formulation. Ideally, such a metric would be convex, allowing the use of traditional optimization techniques. In the following presentation of this evaluation metric, it will be assumed that the observed scene consists of one plane Π of type $T \in \{Building, Road\}$. However, generalization to multiple planes of multiple types is immediate.

For a given FoE candidate, Direct C-Velocity representation of Π can be computed². If the FoE candidate is the actual FoE, the representation of Π will be a perfect parabola. On the other hand, the further the candidate will be from the actual FoE, the less points from Π will accumulate, thus forming a degenerate paraboloid.

It has been exhibited in [14] that the standard deviation of a given plane in the C-Velocity voting space is a function of the distance between the FoE candidate and the actual FoE:

$$\sigma_{\Pi} = d(\|FoE_{candidate} - FoE_{actual}\|) \quad (10)$$

and that this function d is monotonically increasing, Fig. 4 and 5 illustrate this dispersion.

Knowing that the quality of the candidate FoE is directly related to the representation of Π in the C-Velocity voting space, the dispersion of the point cloud representing Π will then be used as an energy function:

$$\epsilon^2 = \sum_{m \in \Pi} (C_{observed} - C_{avg})^2 \quad (11)$$

Where $C_{observed}$ is the observed C-value, and C_{avg} is the average c-value, for a given w . Given the fact that dispersion is convex with respect to the relative position between the candidate and the actual FoE, this metric is also convex.

In this case, the error function is strictly convex with respect to the distance between the candidate and the actual FoE. As such, the problem of localizing the FoE becomes a simple energy minimization problem, for which traditional techniques can be used.

² In the multi-plane case, a separate voting per plane is used in order to eliminate cross-talk perturbation.

Fig. 4. Building plane voting spaces. Left: FoE hypothesis is located at the center of the image ; Right: FoE hypothesis is the one extracted. Parabolas are the best fit ones, horizontal line only indicates the origin of the c-values.

Fig. 5. Normalized dispersion inside the voting space with respect to the distance between the actual and the candidate FoE (synthetic data)

4 Results

Two sets of experiments were conducted. In the first one, the Sivic simulator [16] was used in order to accurately simulate a monocular system, attached to a car. In a second part, real image sequences were used. Those images are from the french ANR project *Logiciel d’Observation des Vulnérables (LoVE)*.

The experiments were conducted using optical flow provided by the FOLKI [17] method.

The FoE search was conducted using a simple Gradient-Descent approach.

4.1 Simulated Data

The Sivic simulator was used to generate a sequence of 250 640x480 images pairs, presenting a urban environment with moderate traffic. Along this sequence, the trajectory of the vehicle was mostly translational.

Fig. 3 presents one image extracted from this sequence, as well as the planes (overlays) extracted using the procedure described in 3.1.

For the same image, Fig. 6 presents the computed optical flow and the extracted Focus of Expansion, it is important to note that the extracted FoE lies on the line of horizon, which is consistent with the known motion of the camera. More specifically, for this particular image, the error between the estimated Focus of Expansion and the actual FoE was 2.2 pixels.

Over this sequence, the maximum positioning error was 15.6 pixels and the average positioning error was 5.2 pixels. This error can appear large, however,

with the simulated optical system (a focal length of $10mm$ and a pixel size of $10\mu m$), an error of one pixel in the localization of the FoE translates to an error of 10^{-3} on the ratio $\frac{\|T\|}{T_z}$, with: $T = \begin{bmatrix} T_X \\ T_Y \end{bmatrix}$, for an order of magnitude, this represents a lateral translation of $0.6mm$ for a car moving at $50km.h^{-1}$.

The measured error can then be translated to an error on the estimation of the ratio $\frac{\|T\|}{T_z}$ of 0.8%.

As a comparison, for the same image sequence a standard cumulative approach [7] leads to the following results: a maximum positioning error of 14 pixels and an average positioning error of 10.6 pixels, this can be translated to an error of 1.1% on the estimation of the ratio $\frac{\|T\|}{T_z}$.

4.2 Real Data

Unfortunately, for the real data evaluation, ground-truth results are not available. As a consequence, quantitative results can not be produced. However, it is possible to qualitatively assess the validity of those results. For instance, Fig. 4 presents a visualization of the building voting space for an arbitrary FoE candidate, and the extracted FoE. This figure particularly illustrates the geometrical sense of the presented metric. The closer the candidate is to the actual FoE, the more the C-Velocity curve will fit the expected parabola.

Fig. 7, Fig.8 and Fig.9 present computed optical flows along with the extracted Focus of Expansion. Fig. 4 illustrates the building voting space for (respectively), the best extracted FoE and the initial candidate, located at the center of the frame. Particularly, it is interesting to note that, in both cases, the extracted FoE is coherent with the motion of the car. Optical Flow field lines can also be used as a visual cue to evaluate the position of the FoE.

Fig. 6. Image extracted from a Sivic sequence, including the computed Optical Flow and the extracted FoE

Fig. 7. Image Extracted from a real sequence, including the computed Optical Flow and the extracted FoE

Fig. 8. Image Extracted from a different sequence, including the computed Optical Flow and the extracted FoE

Fig. 9. Image Extracted from a different sequence, including the computed Optical Flow and the extracted FoE

5 Discussion – Prospective

Through this work, a method for computing the translational ego-motion of a monocular system has been unveiled. This method rely only on relative norm of the optical flow and so, it is insensitive to noisy estimation of its angular component and to bias on its norm. Furthermore, it uses robust cumulative methods which makes it robust toward punctual noise.

As it stems from a method for extraction scene structure information from motion, it will allow the definition of a single, iterative and cumulative framework for scene structure and ego-motion estimation in future developments.

Preliminary results are encouraging, however, some effort should be put into the study of the impact of noise as well as on the development of an absolute evaluation method for real data sets. Furthermore, rotations are currently neglected and should be taken into account.

Finally, the presented system could benefit from a fine analysis of the dispersion in the C-Velocity spaces, due to a shift of the FoE. This will be presented in future works.

Acknowledgements. The authors would like to acknowledge the french cluster for research in new technologies Digiteo for its funding and support to their projects.

References

1. Bouchafa, S., Zavidovique, B.: C-Velocity: A Cumulative Frame to Segment Objects From Ego-Motion. *Pattern Recognition and Image Analysis* 19, 583–590 (2009)
2. McCarthy, C., Barnes, N., Mahony, R.: A Robust Docking Strategy for a Mobile Robot using Flow Field Divergence. *IEEE Transactions on Robotics* 24, 832–842 (2008)

3. Ancona, N., Poggio, T.: Optical flow from 1d correlation: Application to a simple time-to-crash detector. In: Proceedings of the International Conference on Computer Vision, pp. 209–214 (1993)
4. Fukuchi, M., Tsuchiya, T., Koch, C.: The focus of expansion in optical flow fields acts as a strong cue for visual attention. *Journal of Vision* 9(8), article 137 (2009)
5. Prazdny, K.: Motion and Structure from Optical Flow. In: Proceedings of the Sixth International Joint Conference on Artificial Intelligence (1979)
6. Roach, J.W., Aggarwal, J.K.: Determining the Movement of Objects from a Sequence of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2, 554–562 (1980)
7. Suhr, J.K., Jung, H.G., Bae, K., Kim, J.: Outlier rejection for cameras on intelligent vehicles. *Pattern Recognition Letters* 29, 828–840 (2008)
8. Sazbon, D., Rotstein, H., Rivlin, E.: Finding the focus of expansion and estimating range using optical flow images and a matched filter. *Machine Vision and Applications* 15, 229–236 (2004)
9. Hummel, R., Sundareshwaran, V.: Motion-parameter Estimation from Global Flow Field Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(5), 459–476 (1993)
10. Wu, F.C., Wang, L., Hu, Z.Y.: FOE Estimation: Can Image Measurement Errors Be Totally "Corrected" by the Geometric Method? *Pattern Recognition* 40(7), 1971–1980 (2007)
11. Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., Szeliski, R.: A Database and Evaluation Methodology for Optical Flow. In: *IEEE International Conference on Computer Vision*, pp. 1–8 (2007)
12. Rodriguez, S.A., Frémont, F.V., Bonnifait, P.: Extrinsic Calibration Between a Multi-Layer LIDAR and a Camera. In: *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 214–219 (2008)
13. Iocchi, L., Konolige, K., Bajracharya, M.: Visually Realistic Mapping of a Planar Environment with Stereo. In: *Experimental Robotics VII. LNCIS*, vol. 271, pp. 521–532 (2001)
14. Bouchafa, S., Patri, A., Zavidovique, B.: Efficient Plane Detection From a Single Moving Camera. In: *International Conference on Image Processing*, pp. 3493–3496 (2009)
15. Labayrade, R., Aubert, D., Tarel, J.-P.: Real-Time Obstacle Detection on Non-Flat Road Geometry Through V-Disparity Representation. In: *Intelligent Vehicles Symposium*, pp. 646–651 (2002)
16. Gruyer, D., Royere, C., du Lac, N., Michel, G., Blosseville, J.-M.: SiVIC and RTMaps, interconnected platforms for the conception and the evaluation of driving assistance systems. In: *IEEE Conference on Intelligent Transportation Systems* (2006)
17. Le Besnerais, G., Champagnat, F.: Dense Optical Flow by Iterative Local Window Registration. In: *IEEE International Conference on Image Processing*, vol. 1, pp. 137–140 (2005)