IPOL - CLG Optical Flow

Jorge Jara, José Delpiano & Steffen Härtel,
SCIAN-Lab, DCC, Biomedical Neuroscience Institute, University of the Andes,
University of Chile

June 7, 2012

1 Overview

Optical flow (OF) approaches calculate vector fields which determine the apparent velocities of objects in time varying image sequences. OF was introduced in 1981 by Horn and Schunck (HS) as "...the distribution of apparent velocities of movement of brightness patterns in an image" [5]. The idea contains two basic assumptions: "brightness value constancy" and "smooth flow of the intensity values" between two successive images. At about the same time, Lucas and Kanade (LK) presented a method for motion estimation between images, considering constant motion patterns for image patches [7]. While the HS model is intended to solve the problem for a non-null motion field over the entire image -a *global* approach- the LK model can produce homogeneus piece-wise motion field "patches" -as a *local* approach- allowing null motion regions. Several variations of the original HS- and LK-OF approaches have been published. The presented algorithm is known as the combined local global (CLG) method of Bruhn et al. [2], encompassing properties of both the global HS- and the local LK-OF models. The CLG-OF aims to improve the accuracy of the OF field for small-scale variations while retaining the HS-OF benefits of dense and smooth vector fields.

An online implementation of this algorithm is available for 2D images, using the numerical scheme proposed in [1]. It must be noted that the algorithm works on grey-scale images. Color images will be converted to single channel (grey-scale) prior to the online OF computation.

2 Description

2.1 Horn-Schunck Global Approach

The HS model assumes "brightness constancy" and a "smooth flow" pattern between two consecutive images. Let $I = I(x, y, t)$, the constancy assumption is expressed as $\partial I/\partial t = 0$. Let $\underline{V}$ the optical flow vector field, $\underline{V} = [u(x, y, t), v(x, y, t), 1]^T$. The HS-OF is defined from a first-order Taylor expansion

$$I(x + u, y + v, t + \Delta t) - I(x, y, t) = 0 \implies I_x u + I_y v + I_t = 0 \tag{1}$$

Integrating over the image domain Ω, an "energy functional" is defined as

$$E_{HS}(\underline{V}) = \int_\Omega (I_x u + I_y v + I_t)^2 + \alpha(|\nabla u|^2 + |\nabla v|^2)dxdy \tag{2}$$

The **global regularization coefficient** $\alpha > 0$ is a smoothing factor for the OF field. Higher values produce more homogeneus fields, while lower values allow more variating displacement vectors in a given image region. It must be noted that this coefficient is denoted by α^2 in the HS-OF article [5], instead of α in the CLG-OF article [2].

2.2 Lucas-Kanade Local Approach

The LK approach assumes that the flow is constant in a neighbourhood of size ρ, computing the optical flow for all the pixels in that neighbourhood, by the least squares criterion

$$E_{LK}(u,v) = K_\rho * (I_x u + I_y v + I_t)^2 \tag{3}$$

A minimum (u,v) for E_{LK} satisfying $\partial_u E_{LK} = 0$ and $\partial_v E_{LK} = 0$ yields the following system of equations

$$\begin{bmatrix} K_\rho * (I_x^2) & K_\rho * (I_x I_y) \\ K_\rho * (I_x I_y) & K_\rho * (I_y^2) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -K_\rho * (I_x I_t) \\ -K_\rho * (I_y I_t) \end{bmatrix} \tag{4}$$

If the image gradients are not zero, the matrix for the system is invertible and a solution can be computed.

2.3 Bruhn et al. Combined Local-Global Approach

Bruhn et al. [2] defined the CLG-OF by an integral functional that encompasses both the HS and the LK approaches. This combined approach aims to produce dense flow fields (characteristic of the global methods) which are robust against noise, by employing smoothing terms based on both the global and local approaches.

Let

$$\nabla_3 I = [I_x, I_y, I_t]^T$$

The HS-OF energy functional can be written as

$$E_{HS}(\underline{V}) = \int_\Omega (\underline{V}^T(\nabla_3 I \nabla_3 I^T)\underline{V} + \alpha(|\nabla u|^2 + |\nabla v|^2))dxdy \tag{5}$$

The CLG-OF energy functional introduces a smoothing term in $\nabla_3 I$. The corresponding energy functional is defined as

$$E_{CLG}(\underline{V}) = \int_\Omega (\underline{V}^T J_\rho(\nabla_3 I)\underline{V} + \alpha(|\nabla u|^2 + |\nabla v|^2))dxdy \tag{6}$$

with

$$J_\rho(\nabla_3 I) = G_\rho \otimes (\nabla_3 I \nabla_3 I^T)$$

J_ρ acts as a **local spatio-temporal derivative smoothing** term, defined as a convolution $\otimes$ between a bi-dimensional Gaussian kernel G_ρ, with standard deviation ρ, with the matrix $(\nabla_3 I \nabla_3 I^T)$. If $\rho = 0$ no local smoothing occurs, making the CLG-OF functional equal to the HS-OF. If $\alpha = 0$ the functional becomes equivalent to the LK-OF model.

2.3.1 Numerical solution

The minimum of the energy functional has to satisfy the Euler-Lagrange equation, in the form of a system of partial differential equations

$$\alpha \Delta u - (J_{11}u + J_{12}v + J_{13}) = 0 \tag{7}$$
$$\alpha \Delta v - (J_{21}v + J_{22}v + J_{23}) = 0 \tag{8}$$

with J_{ik} denoting the elements of the matrix J_ρ. Neumann boundary conditions are assumed, given by

$$\partial_n u = 0, \partial_n v = 0 \tag{9}$$

Although the CLG-OF can be equivalent to the HS-OF depending on the parameter values, different discretization schemes were used for both models in their original versions. This can lead to slightly different results when both methods are applied to the same set of images (see for example [3, 6] which evaluate OF for fluorescent objects in microscopy images).

The discrete derivatives in space and time (used to compute J_ρ) are computed as

$$I_x \approx I \otimes \frac{1}{12h}[1 \ \ -8 \ 0 \ 8 \ \ -1]_x|_{x_i,y_j,t_k} \tag{10}$$

$$I_y \approx I \otimes \frac{1}{12h}[1 \ \ -8 \ 0 \ 8 \ \ -1]_y|_{x_i,y_j,t_k} \tag{11}$$

$$I_t \approx I(x_i, y_j, t_{k+1}) - I(x_i, y_j, t_k) \tag{12}$$

where h is a step size of the discretized domain (a single pixel can be taken as $h = 1$, for instance, and $A_l|_{x_i,y_j,t_k}$ denotes the mask vector A applied in the l axis at image position (x_i, y_j, t_k) (the central element of the mask corresponds to the image pixel (x_i, y_j, t_k)).

J_ρ is computed as follows: (i) $J_0 = \nabla_3 I \nabla_3 I^T$ is computed, (ii) if $\rho = 0$ then $J_\rho = J_0$, otherwise (iii) a square gaussian matrix G_ρ is defined with standard deviation ρ, and (iv) $J_\rho = J_0 \otimes G_\rho$.

The laplacian is computed by a convolution between the OF components and a kernel matrix M as

$$\Delta u_{x_i,y_j,t_k} \approx U \otimes M|_{x_i,y_j,t_k} \Delta v_{x_i,y_j,t_k} \approx V \otimes M|_{x_i,y_j,t_k} \tag{13}$$

with

$$M = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

2.4 Iterative scheme

The first step in the CLG-OF computation is a gaussian smoothing (optional) of the input images. Upon the result, the numerical solution is computed using a Gauss-Seidel iterative scheme, according to [1]. Given the system $Ax = b$, the matrix A is decomposed in the form $A = D - L - U$, with L a diagonal matrix and L, U lower and upper triangular matrices. Then, the k-th Gauss-Seidel iteration for x is given by $x^{k+1} = (D - L)^{-1}(Ux^k + b)$. From a given initial value x^0, the equation is solved iteratively until a convergence criterion is reached. In this case, two equations must be solved for the OF field components, (u, v) as described next. For these equations the subscript notation is changed from two indexes (e.g. u_{ij}) to one index (u_i), in order to be consistent with the original article.

Let U and V the matrices storing the x and y components of the OF field at a given position i and iteration k, the values of the OF field (u_i, v_i) for the next iteration are computed as

$$u_i^{k+1} = \frac{\displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} \left(\sum_{j \in N_l^-(i)} u_j^{k+1} + \sum_{j \in N_l^+(i)} u_j^k \right) - (J_{12i} v_i^k + J_{13i})}{\displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} |N_l(i)| + J_{11i}} \tag{14}$$

$$v_i^{k+1} = \frac{\displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} \left(\sum_{j \in N_l^-(i)} v_j^{k+1} + \sum_{j \in N_l^+(i)} v_j^k \right) - (J_{21i} u_i^{k+1} + J_{23i})}{\displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} |N_l(i)| + J_{22i}} \tag{15}$$

where $N_l(i)$ denotes the neighbors of pixel i in direction of axis l belonging to Ω, making

$$N_l^+(i) = \{j \in N_l(i) | j > i\}$$
$$N_l^-(i) = \{j \in N_l(i) | j < i\}$$

At this point, a variant of the Gauss-Seidel method is used. The so-called *pointwise coupled Gauss-Seidel* scheme update the OF values for each pixel synchronously [1, 4]. Let the vector $w_i^{k+1} = (u_i^{k+1}, v_i^{k+1})$ at pixel i, the iteration becomes

$$w_i^{k+1} = M_i^{-1} g_i^{k+1/2} \tag{16}$$

with

$$M_i = \begin{bmatrix} \displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} |N_l(i)| + J_{11i} & J_{12i} \\ J_{21i} & \displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} |N_l(i)| + J_{22i} \end{bmatrix}$$

and

$$g_i^{k+1/2} = \begin{bmatrix} \displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} \left(\sum_{j \in N_l^-(i)} u_j^{k+1} + \sum_{j \in N_l^+(i)} u_j^k \right) - J_{13i} \\ \displaystyle\sum_{l=1}^{2} \frac{\alpha}{h_l^2} \left(\sum_{j \in N_l^-(i)} v_j^{k+1} + \sum_{j \in N_l^+(i)} v_j^k + \right) - J_{23i} \end{bmatrix}$$

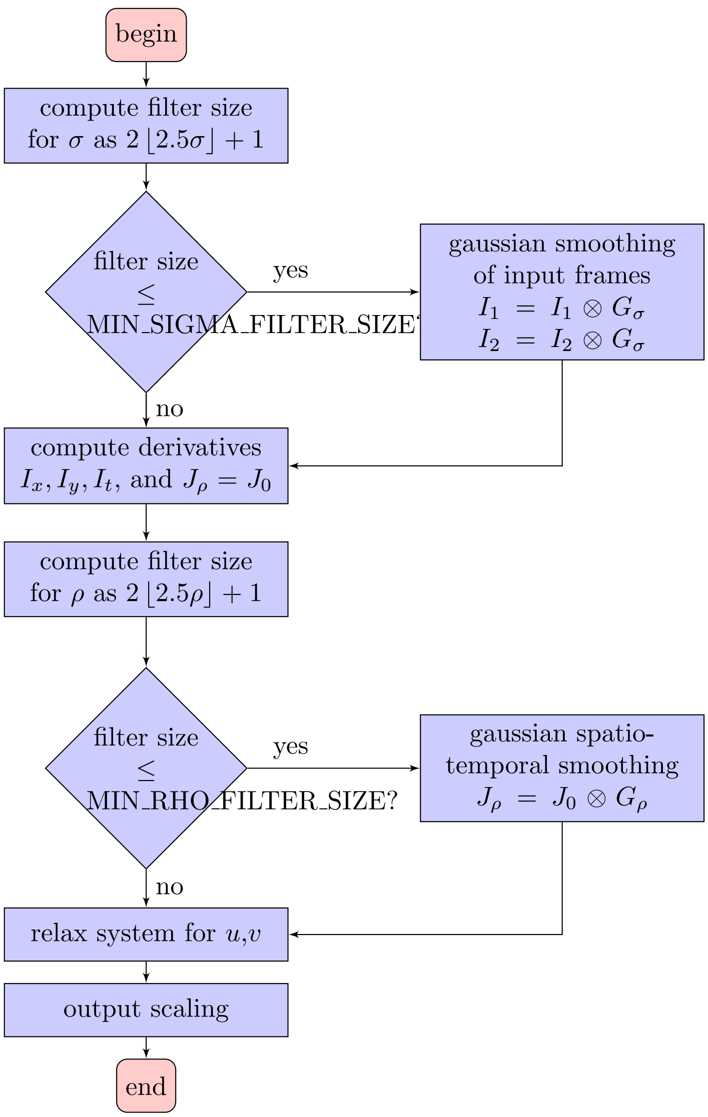
Although the computation for u_i^{k+1} and v_i^{k+1} can be performed sequentially (i.e. one iteration loop for u, then another for v), an alternating computation for both can prevent problems for small α values that make the term $w^T J_\rho(\nabla_3 f) w$ dominate in the solution. In addition, this scheme can be used for faster and more complex solver algorithms such as multigrid, as described in [1].

3 Coded algorithm

The main algorithm is implemented according to the flow diagram that follows.

Input

- I_1, I_2: the input images for the OF computation.
- $\alpha > 0$: the global regularization coefficient value.
- $\rho > 0$: the local derivative regularization coefficient value.
- $\sigma \geq 0$: the standard deviation value for gaussian smoothing.
- *iterations*: the number of relaxation iterations for the OF vector field.



3.1 Constants

- $JROWS = 3$ Number of rows of the J_ρ matrix associated to a given image position.
- $JCOLS = 3$ Number of columns of the J_ρ matrix associated to a given image position.
- $MIN\_FILTER\_SIZE\_SIGMA = 3$ Minimum filter size required to perform gaussian smoothing of the input images/frames. If the computed filter size is lower than this value, no smoothing is performed.
- $MIN\_FILTER\_SIZE\_RHO = 3$ Minimum filter size required to perform the gaussian smoothing of the J_ρ matrix (local spatio-temporal derivative smoothing). If the computed filter size is lower than this value, no smoothing is performed.
- $EPS = 1 \cdot 10^{-12}$ Precision threshold (epsilon) value for numerical computations. Used by the relaxation funcion.

3.2 Functions description

This section summarizes the purpose and parameters of the implemented functions. Pointer variables to 1- and 2-dimensional arrays are denoted with the prefixes * and **, respectively.

- $calcCLG\_OF(*prevImage, *currImage, *u, *v, nRows, nCols, numIterations, alpha, rho, sigma)$. Entry point for the CLG-OF computation. Implements the algorithm as summarized in the flow diagram, and performs the required memory allocations and deallocations, data formatting and initializations. $prevImage$ and $currImage$ are pointers to the two input images for the OF computation. u, v point to the vertical and horizontal OF components respectively (output). $nCols$ and $nRows$ are input values that specify the size of the input images and output OF. $numIterations, alpha, rho$ and $sigma$ are input values for the number of iterations, global smoothing coefficient, local spatio-temporal smoothing coefficient and gaussian image smoothing parameters, respectively.
- $matrixSmooth(**matrix, nRows, nCols, filterSize, filterSigma)$. Performs a gaussian smoothing for a given input matrix of size $nRows \times nCols$, using a square kernel matrix (the filter) parametrized by its size and standard deviation sigma. This function is used by $calcCLG\_OF$, according to the values of $sigma$ and rho, to smooth the input images and the derivatives matrix, respectively.
- $computeDerivatives(**image1, **image2, nRows, nCols, **J)$. Performs and store discrete derivative computations for the two input images, a "previous" ($image1$) and a "current" ($image2$). The size of each frame is $nRows \times nCols$. The computed derivatives are stored in a matrix of size $nRows \times nCols \times JROWS \times JCOLS$, with J as the pointer to this matrix. For a given pixel $[i, j]$, the discrete derivatives as stored as depicted:

$$J[1][1][i][j] = I_x[i][j] * I_x[i][j]$$
$$J[1][2][i][j] = I_x[i][j] * I_y[i][j]$$
$$J[1][3][i][j] = I_x[i][j] * I_t[i][j]$$
$$J[2][2][i][j] = I_y[i][j] * I_y[i][j]$$
$$J[2][3][i][j] = I_y[i][j] * I_t[i][j]$$
$$J[3][3][i][j] = I_t[i][j] * I_t[i][j]$$

  Since $J[][][i][j]$ is a symmetric matrix with respect to its diagonal, its lower triangular part is not stored.

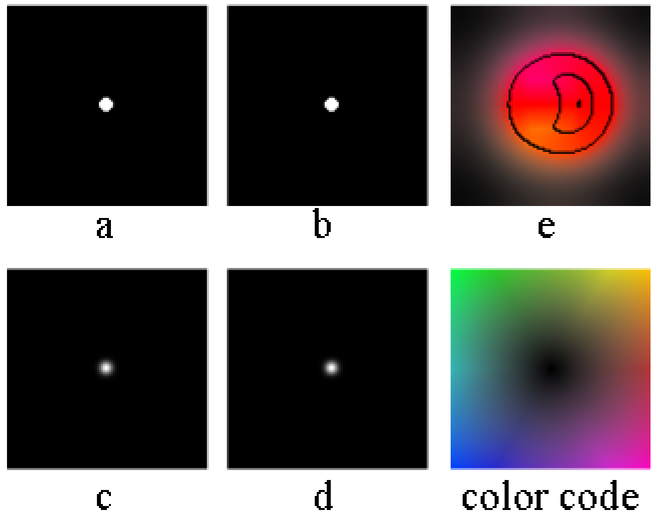


Figure 1: Moving point source simulating a fluorescent signal in a confocal microscopy image, and its corresponding CLG-OF vector field. **a,b**: first and second time frames. **c,d**: PSF-convolved images from a,b according to the theoretical confocal microscope PSF. **e**: CLG-OF vector field computed with $\alpha = 200$, $\rho = 5$, $\sigma = 0$, and 500 iterations. The OF vector directions are colored according to the color code. Pixel size corresponds to 107nm in x and y. The displacement of the signal between the two frames is 3 pixels (321nm).

- $relax(**u, **v, **J, nRows, nCols, alpha)$. Pointwise coupled Gauss-Seidel relaxation method for solving the CLG-OF equations. Each call to the function updates the value of the OF vector field according to equation 16, using Cramer's rule. The horizontal and vertical OF components are u, v with size $nRows \times nCols$, $alpha$ denotes the optical flow global smoothing coefficient α, and J is the pointer to the arrays storing the computed (and possibly smoothed) derivatives. Boundary conditions are used according to equation 9.
- $lin2by2det(a, b, c, d)$. Utility function used by $relax$. Computates the determinant of a given 2×2 matrix. Let $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, the determinant d is computed as $d = ad - bc$.

4 Examples

The following figures are examples of the CLG-OF implementation. Our first example is a model structure observed by fluorescence microscopy, corresponding to a model of a point signal from fluorescent proteins. The signal was convolved with a theoretical point spread function (PSF) of a system with a 60x water objective, and wavelenghts of 543/560 nm for excitation/emmision (a PSF acts as a smoothing function that blurries objects. Details of simulation and tests in [3]). A representative OF field is shown in fig. 1.

Figures 2-5 illustrate the effects of different values for α and ρ in the resulting CLG-OF.

References

[1] Andrés Bruhn, Joachim Weickert, Christian Feddern, Timo Kohlberger, and Christoph Schnörr. Variational optical flow computation in real time. *IEEE Transactions on Image Processing*, 14(5):608–615, 2005.

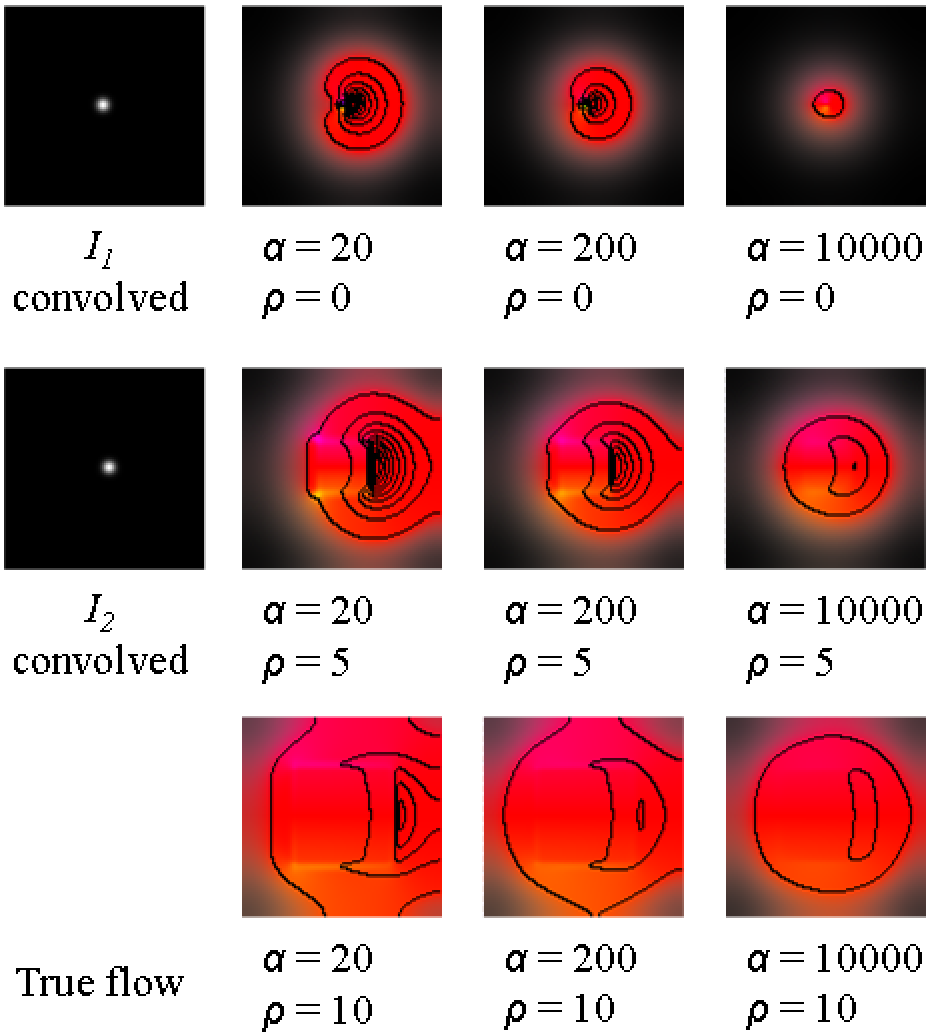


Figure 2: Different parameter values for CLG-OF and their corresponding OF fields for the baboon image, with an horizontal displacement of one pixel. I_1,I_2 are the input time frames.

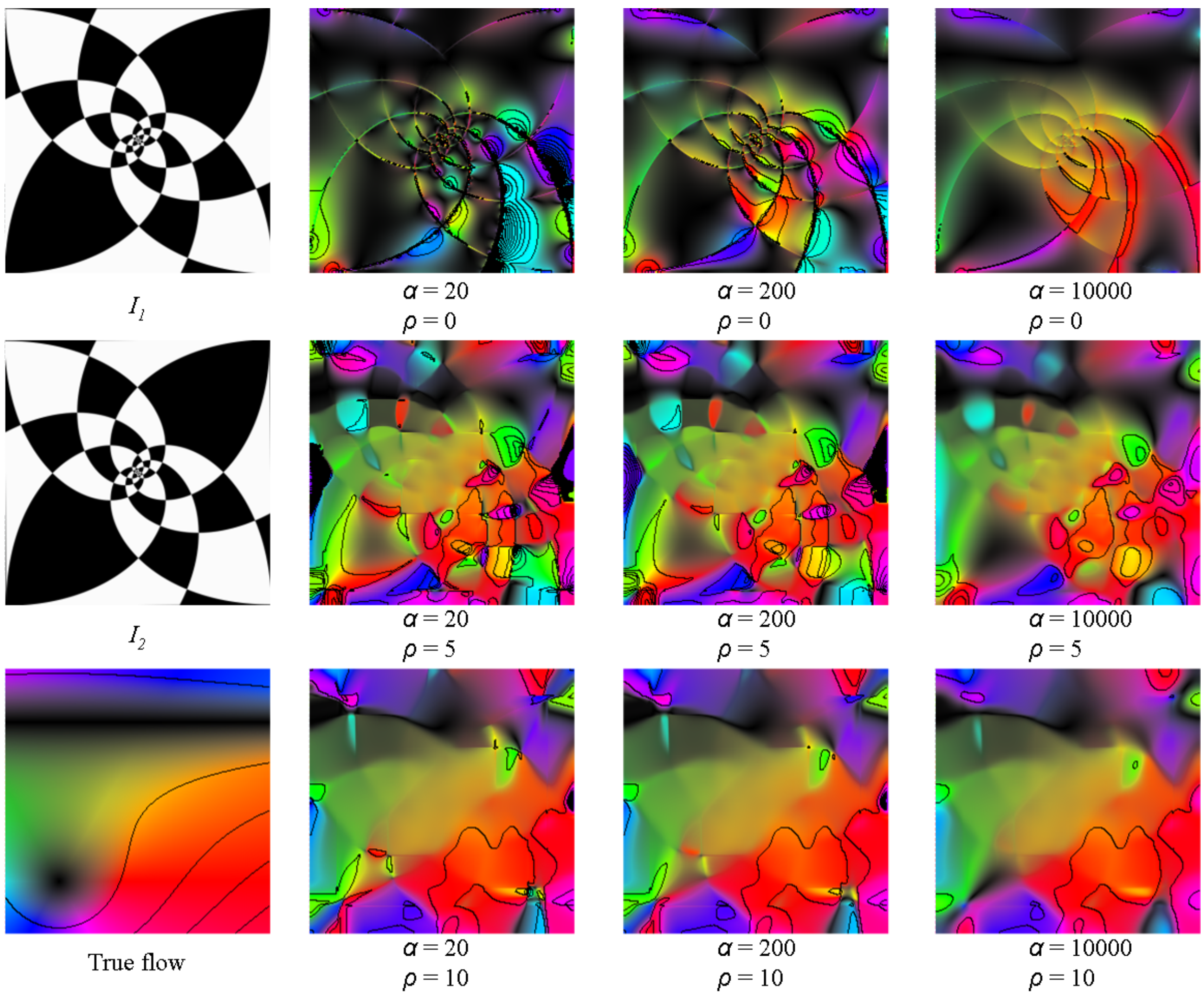


Figure 3: Different parameter values for CLG-OF and their corresponding OF fields for the spiral image, with the homography transformation. I_1,I_2 are the input time frames.

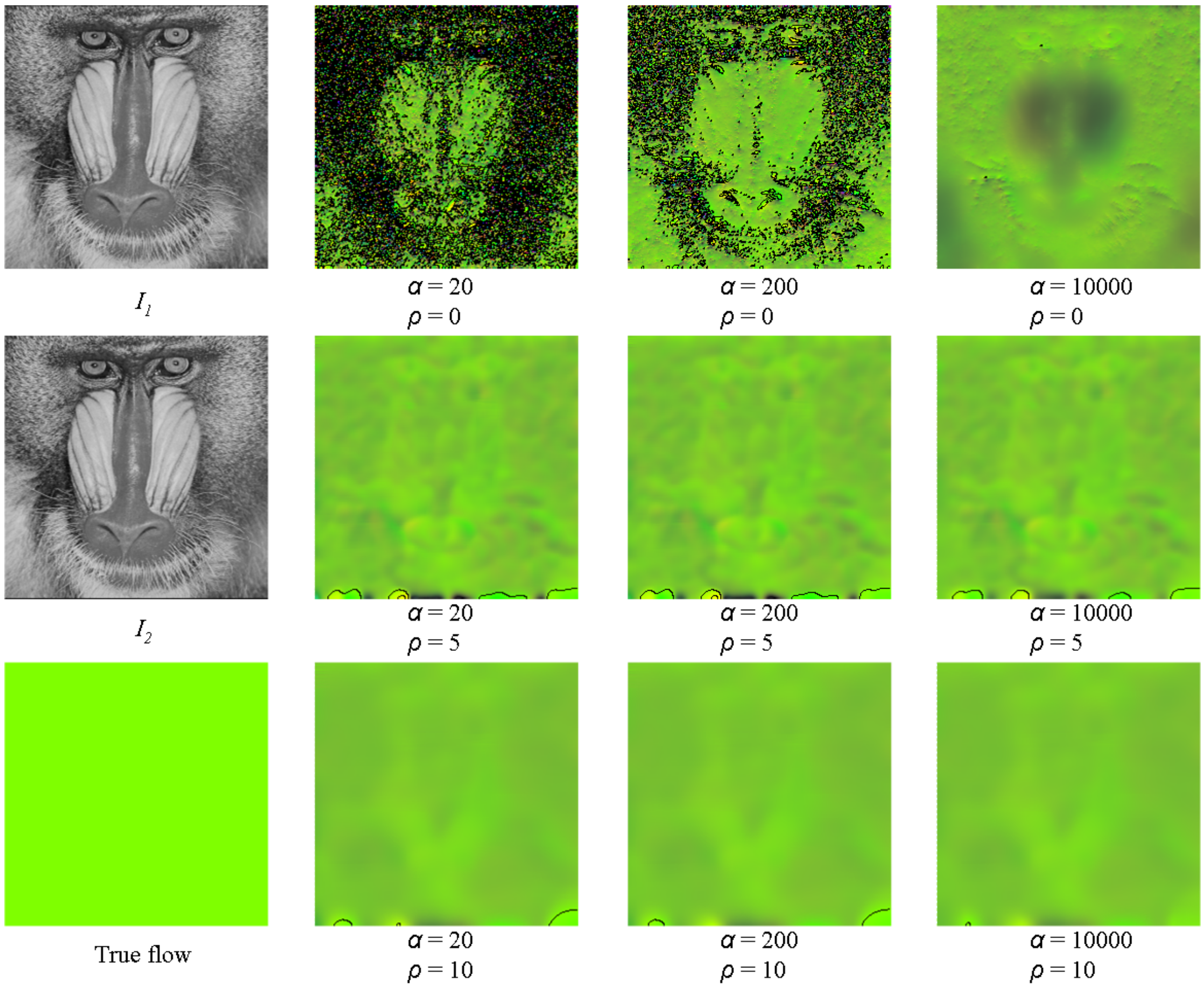


Figure 4: Different parameter values for CLG-OF and their corresponding OF fields for the baboon image, with an horizontal displacement of one pixel. I_1,I_2 are the input time frames.

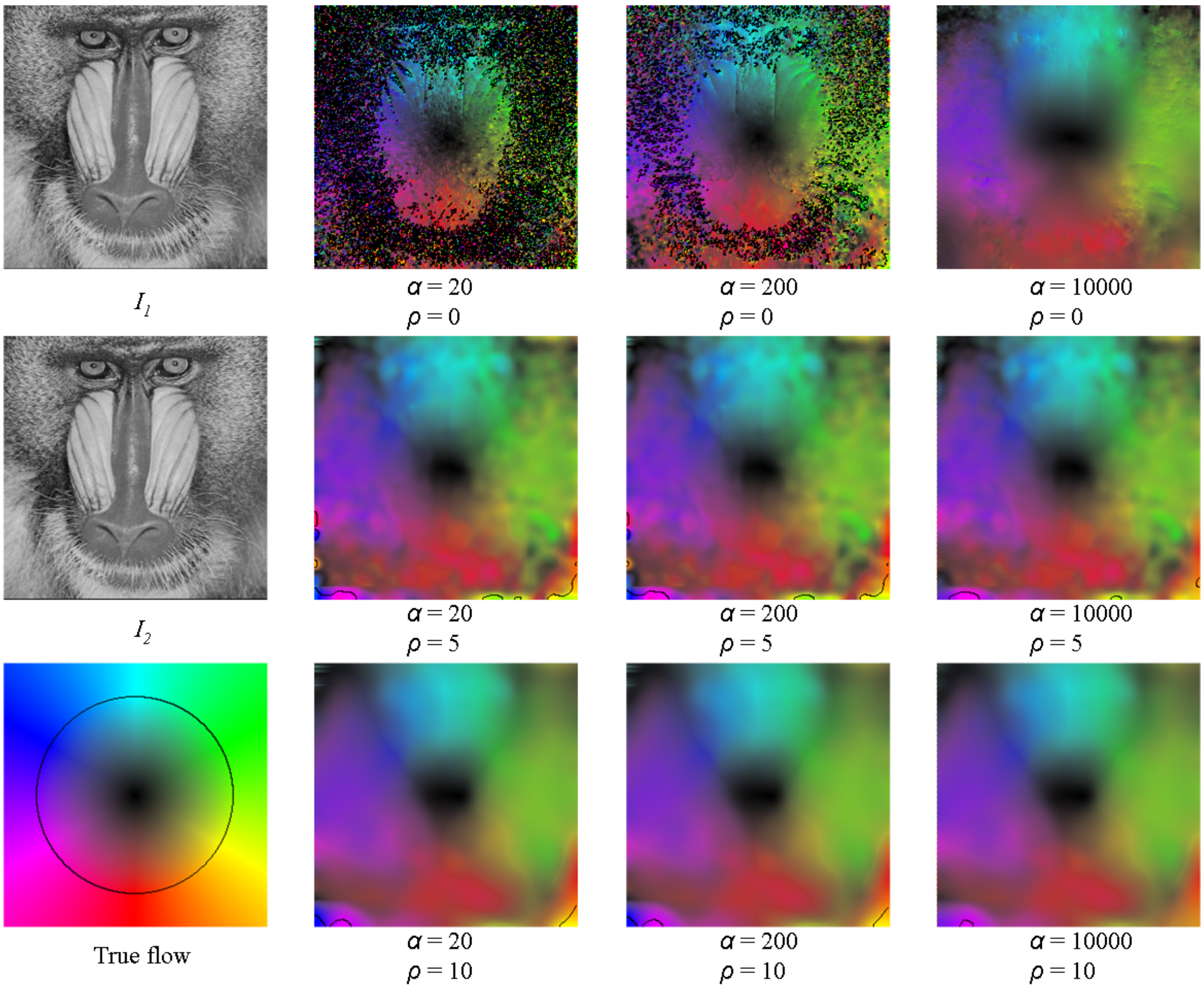


Figure 5: Different parameter values for CLG-OF and their corresponding OF fields for the baboon rotation images. **a,b**: first and second time frames. **c-g**: CLG-OF fields computed for different values of α and ρ.

[2] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Combining the advantages of local and global optic flow methods. In *DAGM-Symposium*, pages 454–462, 2002.

[3] José Delpiano, Jorge Jara, Jan Scheer, Omar A. Ramírez, Javier Ruiz del Solar, and Steffen Härtel. Performance of optical flow techniques for motion analysis of fluorescent point signals in confocal microscopy. *Machine Vision and Applications. Accepted*, pages 1–15, 2011. DOI 10.1007/s00138-011-0362-8.

[4] Wolfgang Hackbusch. *Applied Mathematical Sciences*. Springer, New York, 1993.

[5] Berthold K P Horn and Brian G Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[6] Jan Hubený, Vladimír Ulman, and Pavel Matula. Estimating large local motion in live-cell imaging using variational optical flow. In *VISAPP 2007 Second International Conference On Computer Vision Theory And Applications, Proceedings*, pages 542–548, Barcelona, Spain, 2007. INSTICC - Institute for Systems and Technologies of Information, Control and Communication.

[7] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. 7th Int. Conf. Artificial Intelligence*, pages 674–679, 1981.