Image processing with variational approaches and Partial Differential Equations
Practice 3: Optical flow

Nicolas Papadakis and Jean-François Aujol

IMB, Université Bordeaux 1
351 Cours de la libération, 33405 Talence Cedex, FRANCE

Email : nicolas.papadakis@math.u-bordeaux.fr

1. Introduction

意思说只有原来的像素的移动，并没有“新”的像素进来

The objective of this practice is to implement algorithms dedicated to the estimation of optical flow between successive frames of a video $I(x, y, t)$ defined on a constant $2D$ pixel domain $(x, y) \in \Omega$ for different times $t = 1 \cdots N$. We fill focus on variational methods derived from the seminal work of Horn and Schunck [2]. Assuming that the pixels intensities do not change over time, the optical flow problem between two successive images $I(x, y, t)$ and $I(x, y, t+1)$ can be stated as the estimation of a $2D$ motion field (u, v) so that:

像素强度

$$I(x + u(x, y), y + v(x, y), t + 1) = I(x, y, t), \tag{1.1}$$

where $u(x, y)$ and $v(x, y)$ stand for the horizontal and vertical components of the motion field and are defined over Ω.

This is illustrated in the following figure:

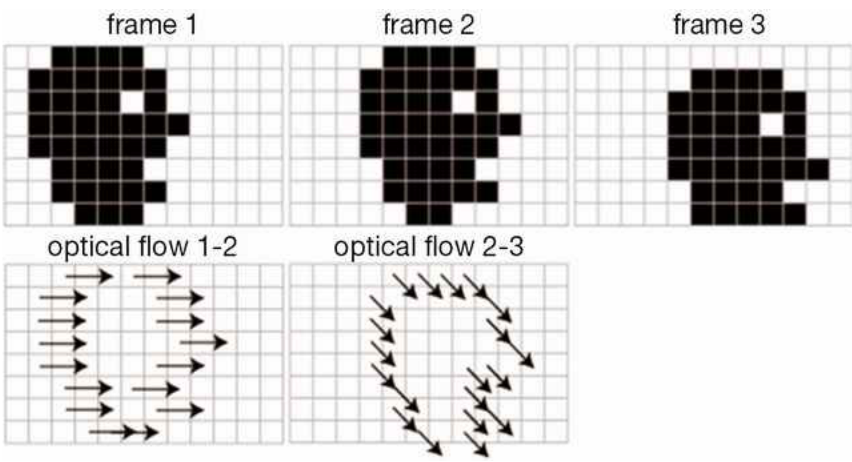


Illustration [3] of motion field $w = (u, v)$ between successive frames.

Relation (1.1) is non linear in (u, v). Assuming that the motion amplitude is small, a linearization can be performed to obtain:

只是用了一阶泰勒展开

$$I(x+u, y+v, t+1) \approx I(x, y, t) + \partial_x I(x, y, t)u(x, y) + \partial_y I(x, y, t)v(x, y) + \partial_t I(x, y, t), \quad (1.2)$$

Considering the partial temporal derivative of the image $\partial_t I(x, t) = I(x, y, t+1) - I(x, y, t)$ and using relation (1.1), we obtain:

$$\partial_t I + \partial_x I u + \partial_y I v = \partial_t I + \nabla I \cdot w \approx 0, \quad (1.3)$$

where $w = (u, v)$ is the motion field to estimate. This constraint, that can also be derived from $\frac{dI}{dt} = 0$, is known as the optical flow constraint equation. Hence, the data term to minimize for the optical flow estimation can be written as:

$$||\partial_t I + \nabla I \cdot w||^2.$$

By minimizing this squared L_2 distance, we are looking for a motion field w that register the image $I(x, y, t+1)$ with respect to the image $I(x, y, t)$.

这个最小二乘问题毫无疑问是ill-posed，因为未知数有2*omega，而方程个数只有Omega个

This problem is nevertheless ill-posed, since two unknowns u and v of total size $2|\Omega|$ have to be estimated from only $|\Omega|$ constraints. As a consequence, Horn and Schunck proposed to add some regularity constraints on the motion components in order to close the problem. Assuming that the underlying motion field is smooth, they consider the Tikhonov regularization:

$$||\nabla w||^2 = ||\nabla u||^2 + ||\nabla v||^2.$$

每个病态问题，我们都用一个regularization term去处理

||gradient(w)||这种形式的regularization term就是tikhonov regularization

The final convex functional to minimize thus reads:

$$J(u, v) = \lambda ||\partial_t I + \partial_x I u + \partial_y I v||^2 + \left(||\nabla u||^2 + ||\nabla v||^2\right), \quad (1.4)$$

where $\lambda > 0$ is the regularization parameter.

2. Horn and Schunck model for color images

We now consider the optical flow estimation between color images I in the RGB space, we will refer to the index $c = 1, 2, 3$ for each color channel. The problem thus becomes:

$$J(u, v) = \lambda \sum_{c=1}^{3} ||\partial_t I^c + \partial_x I^c u + \partial_y I^c v||^2 + ||\nabla u||^2 + ||\nabla v||^2, \quad (2.1)$$

In order to minimize the convex and differentiable functional (2.1), we rely on the simple gradient descent algorithm, even if more advanced optimization tools could be considered. Initializing $u_0 = v_0 = 0$ and deriving functional (2.1), one obtains, for a time step $\tau > 0$:

$$\begin{cases} u_{k+1} = u_k + \tau(\Delta u_k - \lambda \sum_{c=1}^{3}(\partial_t I^c + \partial_x I^c u_k + \partial_y I^c v_k)\partial_x I^c) \\ v_{k+1} = v_k + \tau(\Delta v_k - \lambda \sum_{c=1}^{3}(\partial_t I^c + \partial_x I^c u_k + \partial_y I^c v_k)\partial_y I^c). \end{cases} \quad (2.2)$$

Implementation

- Load the two images I_1 and I_2 from $\boxed{\text{frame10.png}}$ and $\boxed{\text{frame11.png}}$
- Precompute the gradients $\partial_x I$ and $\partial_y I$ on the image I_1 and the temporal derivative as $\partial_t I = I_2 - I_1$. Be careful that these derivatives are computed on each color channel c.
- Implement algorithm (2.2) and test it for $\lambda = 1/400$ and $\tau = 0.02$ with a large number of iterations. A test $||u_{k+1} - u_k||/||u_k|| + ||v_{k+1} - v_k||/||v_k|| < \epsilon = 10^{-3}$ or 10^{-4} can be considered to check the convergence and stop the iterations. 矩阵范数（但我程序中用的是向量形式所以更好），并且注意有除数的地方要保证除数不为0
- Display/Validation of result:
  - The function $\boxed{\text{flowToColor.m}}$ (that needs $\boxed{\text{computeColor.m}}$) takes u and v as arguments and returns an image (to display) representing the optical flow with standard color convention.
  - The vector field can be displayed on the image with the function $\boxed{\text{drawMotion.m}}$ that takes as inputs the vector field (u, v) and the image I_1.
  - Write a function *Registration.m* that takes as inputs the vector field (u, v) and the second image I_2 and realize the registration: $\tilde{I}_2(x, y) = I_2(x + u, y + v)$ using bilinear interpolation. The obtained image should be closed to I_1. 校准：registration

You should obtain the following results:

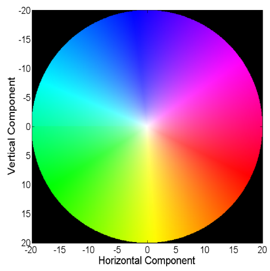
因为在理论推导里面，image domain是连续的（x,y,u,v都是连续的），但在实际中是离散的

I_1

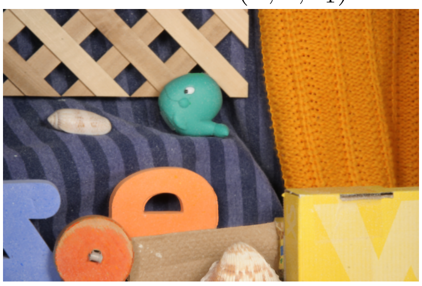
I_2

drawMotion(u, v, I_1)

flowToColor(u, v)



Color convention



$\tilde{I}_2 =$Registration(u, v, I_2)

$|\tilde{I}_2 - I_1|$ 第一个假设是pixel intensity不变，但显然实际中不可能满足，所以I2的校准图片就会在这些地方不等于1

where most of the registration errors $|\tilde{I}_2 - I_1|$ appears in the occluded/desoccluded areas and for the regions with higher motions.

因为这个方法是在运动幅度很小的假设下，所以那些幅度大的区域就会不等于I1

为了可以用之前的假设，必须满足motion amplitude is low，所以借用金字塔。这样大运动幅度会随着层高逐渐降低，从而满足小幅运动

3 Multi-resolution algorithm

并且在本层，上一层传下来的motion vectors已经帮忙本层算好一半了，所以本层只要算剩下的一半，这样的话，剩下那半的amplitude就满足假设了

displacement位移

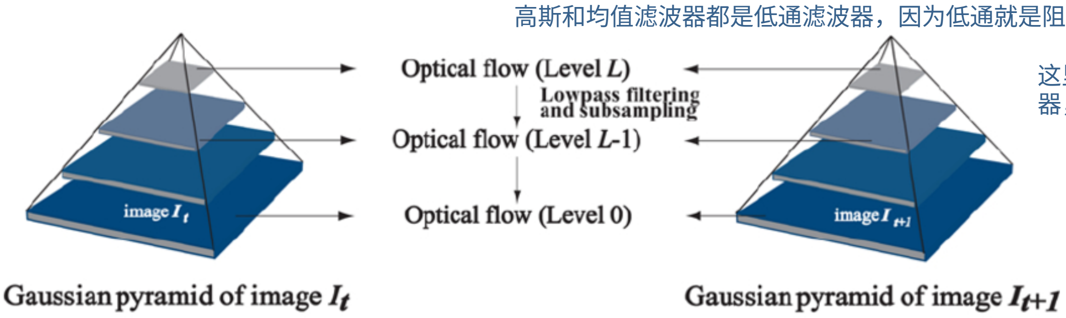
The previous algorithm assumes that the motion amplitude is low. This is not a good model for estimating large displacements. In order to circumvent this limitation, multi-resolution (also known as coarse-to-fine) approaches are commonly used.

The idea is to create a pyramid of down-sampled images I_1 and I_2. We will denote by I^ℓ an image at level ℓ, such that each spatial dimension of the domain of $I^{\ell+1}$ is half the one of I^ℓ and $I^0 = I$, i.e. $|\Omega^\ell| = |\Omega|/4^\ell$ and $\Omega^0 = \Omega$. 就是长，宽都减半，并且此时运动幅度也会同比例一样

Hence, at a low resolution (level $l = 5$ for instance), the number of column and lines of I^ℓ are divided by a factor 2^ℓ with respect to the dimension of I^0. As a consequence, the assumption of small motion amplitude is checked at this scale and the previous algorithm can be used to estimate the motion (u^ℓ, v^ℓ) between images I_1^ℓ and I_2^ℓ.

The multi-resolution framework then consists in estimating the motion $(u^{\ell+1}, v^{\ell+1})$ at a pyramid level $\ell + 1$ and using it at the next level, as illustrated in the following figure.

高斯和均值滤波器都是低通滤波器，因为低通就是阻挡了噪声像素



这里既然写了要用滤波器，就不能是简单地补0

Illustration [5] of the multi-resolution approach.

Hence, we can realize the linearization of the optical flow constraint at the level ℓ around the previously estimated motion field $(u^{\ell+1}, v^{\ell+1})$:

$$\begin{aligned} &I^\ell(x + u^{\ell+1} + u^\ell, y + v^{\ell+1} + v^\ell, t+1) \\ \approx &I^\ell(x + u^{\ell+1}, y + v^{\ell+1}, t) + \partial_x I^\ell(x + u^{\ell+1}, y + v^{\ell+1}, t)u^\ell \\ &+ \partial_y I^\ell(x + u^{\ell+1}, y + v^{\ell+1}, t)v^\ell + \partial_t I^\ell(x + u^{\ell+1}, y + u^{\ell+1}, t), \end{aligned} \tag{3.3}$$

Denoting as $\tilde{I}_2^\ell$ the image I_2^ℓ registered by the motion field $(u^{\ell+1}, v^{\ell+1})$, the linearization can be discretized to obtain the following optical flow constraint:

每一层都不是用I2(l)，而是用I2(l)和上一层的结果生成的registration image I2~(l)去和I1(l)计算

$$\partial_t I^\ell + \partial_x I^\ell u^\ell + \partial_y I^\ell v^\ell = 0,$$

where 注意这里和前面一个部分不一样，用registeration image 和 I2做计算，而不是用I1和registration image做计算

$$\partial_t I^\ell = \tilde{I}_2^\ell - I_1^\ell, \qquad \partial_x I^\ell = \partial_x \tilde{I}_2^\ell, \qquad \partial_y I^\ell = \partial_y \tilde{I}_2^\ell, \tag{3.4}$$

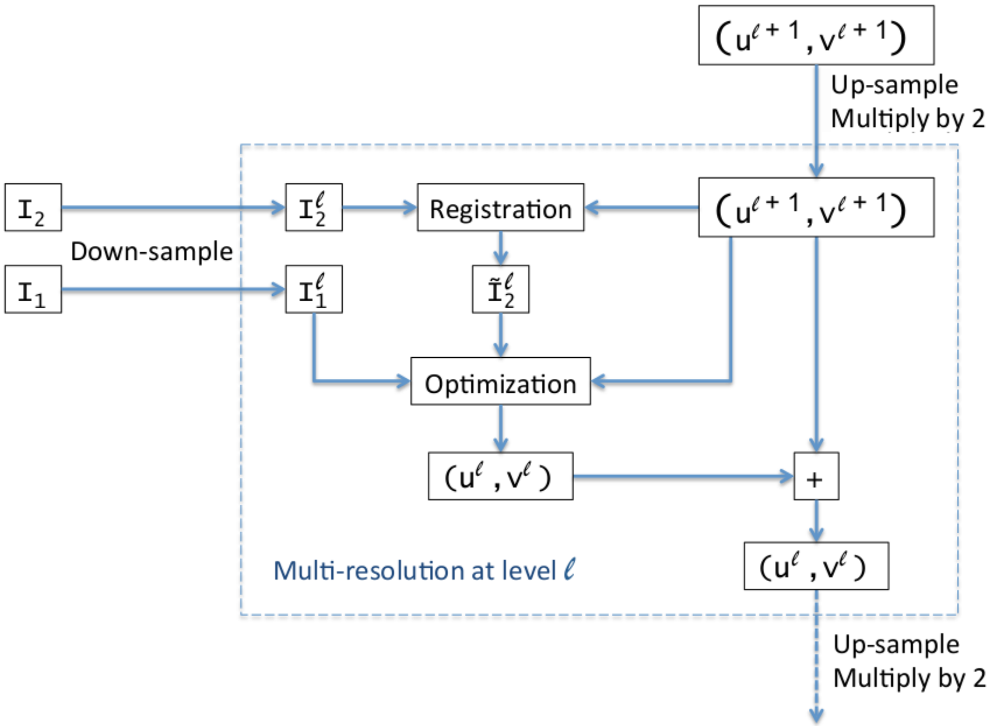
The final convex functional to minimize over (u^ℓ, v^ℓ) at pyramid level ℓ reads:

$$\lambda ||\partial_t I^\ell + \partial_x I^\ell u^\ell + \partial_y I^\ell v^\ell||^2 + ||\nabla(u^\ell + u^{\ell+1})||^2 + ||\nabla(v^\ell + v^{\ell+1})||^2.$$

Notice that the regularization is done over the whole current motion field $(u^\ell + u^{\ell+1}, v^\ell + v^{\ell+1})$. It leads to the following iterative gradient descent for each level ℓ:

$$\begin{cases} u_{k+1}^\ell = u_k^\ell + \tau(\Delta(u_k^\ell + u^{\ell+1}) - \lambda \sum_{c=1}^3 (\partial_t I^{\ell,c} + \partial_x I^{\ell,c} u_k^\ell + \partial_y I^{\ell,c} v_k^\ell)\partial_x I^{\ell,c}) \\ v_{k+1}^\ell = v_k^\ell + \tau(\Delta(v_k^\ell + v^{\ell+1}) - \lambda \sum_{c=1}^3 (\partial_t I^{\ell,c} + \partial_x I^{\ell,c} u_k^\ell + \partial_y I^{\ell,c} v_k^\ell)\partial_y I^{\ell,c}). \end{cases} \tag{3.5}$$

Gathering all elements, the algorithm at level ℓ is described in the following diagram. In the next sections, we will only change the "Optimization" box of this framework.



Implement the multi-resolution algorithm and test it for $L = 6$ with the same parameters than before.

- Inputs: I_1, I_2 and a number of pyramid level L
- Initialize $u^L = v^L = 0$ on the domain Ω^L
- For $\ell = L - 1 \cdots 0$
  - Up-sample $u^{\ell+1}$ and $v^{\ell+1}$ to fit the domain of $\Omega^\ell = |\Omega|/4^\ell$ (that is twice bigger in each dimension than the one of $\Omega^{\ell+1}$). Set $u^{\ell+1} = 2u^{\ell+1}$ and $v^{\ell+1} = 2v^{\ell+1}$ (the motion amplitude is twice bigger on the higher resolution domain Ω^ℓ.
  - Down-sample I_1^ℓ and I_2^ℓ to fit the domain $|\Omega^\ell| = |\Omega|/4^\ell$ from I_1 and I_2

注意不能直接从0层downsample得到l层；不考虑组的概念，也至少得0得到1,1得到2.

  - Register I_2^ℓ with $(u^{\ell+1}, v^{\ell+1})$ to obtain $\tilde{I}_2^\ell$ using your function *Registration.m*
  - Initialize $u_0^\ell = v_0^\ell = 0$. Take $\tau = 0.005$.
  - Realize the gradient descent algorithm (3.5) with (3.4) to obtain (u^ℓ, v^ℓ).
  - Set $(u^\ell, v^\ell) = (u^\ell, v^\ell) + (u^{\ell+1}, v^{\ell+1})$
- Output: (u^0, v^0)

The obtained optical flow is now more accurate as illustrated below.

flowToColor(u, v)

$|\tilde{I}_2 - I_1|$

4. L_1 data term

因为假设说I(x+u,y+v,t+1)-I(x,y,t)几乎为0，所以数据项也很小。但是在occlusion的地方，这两者差很多，所以平方以后会差更多，所以这里称之为 " explode "

In the regions of occlusion/desocclusions, the constraint $I(x+u, y+v, t+1) = I(x, y, t)$ can not be verified so that the norm of the previous data term $||\partial_t I + \partial_x I u + \partial_y I v||^2$ may "explode". In order to be more robust to such outliers, we now consider the following L_1 data term:

$$||\partial_t I + \partial_x I u + \partial_y I v||,$$

金字塔是为了解决第2个假设不成立的情况，这里的1-范数就是为了解决第1个假设不成立的情况

which is still convex in (u, v) but non differentiable, so that gradient descent can not be applied. Hence, we will consider the Forward-Backward algorithm that is dedicated to the minimization of the sum of two convex functionals $F(x) + G(x)$, one of them, say G, being differentiable. This algorithm reads:

$$x_{k+1} = Prox_{\tau F}(x_k - \tau \nabla G(x_k)), \tag{4.1}$$

where the proximity operator is defined as:

$$Prox_{\tau F}(\tilde{x}) = \arg\min_x \frac{||x - \tilde{x}||^2}{2\tau} + F(x).$$

While the step $\tilde{x} = x_k - \tau \nabla G(x_k)$ is an explicit (forward) gradient descent over the function G, the step $Prox_{\tau F}(\tilde{x})$ corresponds to an implicit (backward) descent and is commonly used when the function F is not differentiable.

With our new data term, we can now define the whole optical flow functional to minimize with $F(u, v) = \lambda ||\partial_t I + \partial_x I u + \partial_y I v||$ and $G(u, v) = ||\nabla u||^2 + ||\nabla v||^2$. Notice that for technical reasons, **we will consider from now grayscale images**. As before, the gradient of G is:

$$\nabla_u G(u, v) = -\Delta u, \qquad \nabla_v G(u, v) = -\Delta v. \tag{4.2}$$

Denoting as $\rho(\tilde{u}, \tilde{v}) = \partial_t I + \partial_x I \tilde{u} + \partial_y I \tilde{v}$, the proximity operator of F can be expressed as [6]:

$$Prox_{\tau F}(\tilde{u}, \tilde{v}) = (\tilde{u}, \tilde{v}) + \begin{cases} \tau \lambda \nabla I & if\ \rho(\tilde{u}, \tilde{v}) \leq -\tau\lambda ||\nabla I||^2 \\ -\tau \lambda \nabla I & if\ \rho(\tilde{u}, \tilde{v}) \geq \tau\lambda ||\nabla I||^2 \\ -\rho(\tilde{u}, \tilde{v}) \nabla I / ||\nabla I||^2 & otherwise, \end{cases} \tag{4.3}$$

where $\nabla I = (\partial_x I, \partial_y I)$ is a vector field of the same dimension than (u, v) as we deal with grayscale images. Homework: show relation (4.3). The proximity operator of the color image data term $\sum_c ||\partial_t I^c + \partial I^c_x u + \partial_y I^c v||$ is more complex to compute, see [4] if you are ambitious.

Implement the multi-resolution algorithm for the L_1 data term on grayscale images. Do not forget to convert the color images into grayscale ones, using for instance MATLAB's function *rgb2gray*. With respect to the squared L_2 model, the gradient descent algorithm (3.5), should now be replaced by the Forward-Backward algorithm (4.1) applied to the new problem. This gives, at pyramid level ℓ and iteration $k+1$:

$$\begin{aligned} (\tilde{u}, \tilde{v}) &= (u_k^\ell, v_k^\ell) - \tau \nabla G(u_k^\ell + u^{\ell+1}, v_k^\ell + v^{\ell+1}) \\ (u_{k+1}^\ell, v_{k+1}^\ell) &= Prox_{\tau F}(\tilde{u}, \tilde{v}), \end{aligned} \tag{4.4}$$

using the above relations (4.2) and (4.3). One can take $\tau = 1/8$ and $\lambda = 1/300$.

This new model is more robust to outliers and it also decreases the registration error.

flowToColor(u, v)

$|\tilde{I}_2 - I_1|$

5. Total variation regularization

All the previous models rely on a Tikhonov regularization of the motion field. This leads to smooth motion transitions. In order to promote piece wise constant motion regions and thus sharper transitions, we now consider the Total Variation regularization of the motion field (u, v), as proposed in [6].

The corresponding functional to minimize, still defined for grayscale images, reads:

$$\lambda||\partial_t I + \partial_x I u + \partial_y I v|| + ||\nabla u|| + ||\nabla v||.$$

全部都是1范数

Relying on the dual formulation of the total variation (see Practice 2) and introducing the dual variables $z^u = (z^u_x, z^u_y)$ and $z^v = (z^v_x, z^v_y)$, minimizing the previous functional is equivalent to solve the primal-dual saddle point problem:

$$\min_{u,v} \max_{z^u, z^v} \lambda||\partial_t I + \partial_x I u + \partial_y I v|| + \langle \nabla u, z^u \rangle + \langle \nabla v, z^v \rangle - \iota_B(z^u) - \iota_B(z^v),$$

where ι_B is the characteristic function of the ℓ_2 ball of radius 1, i.e. $\iota_B(z) = 0$ if $||z|| \leq 1$ and $+\infty$ otherwise. To solve this problem, we rely on the primal-dual algorithm of [1] that reads, at level ℓ and iteration $k+1$:

$$\begin{aligned} z^u_{k+1} &= Proj_B\left(z^u_k + \sigma \nabla(u^\ell_k + u^{\ell+1})\right) \\ z^v_{k+1} &= Proj_B\left(z^v_k + \sigma \nabla(v^\ell_k + v^{\ell+1})\right) \\ (u^\ell_{k+1}, v^\ell_{k+1}) &= Prox_{\tau F}(u^\ell_k + \tau div(z^u_{k+1}), v^\ell_k + \tau div(z^v_{k+1})), \end{aligned} \tag{5.1}$$

where $Prox_{\tau F}$ is still defined in (4.3) and the projection onto the ball B is:

$$Proj_B(z) = \begin{cases} z & if\ ||z|| \leq 1 \\ z/||z|| & otherwise. \end{cases}$$

Implement the multi-resolution algorithm for the L_1 data term and Totaal Variation regularization on grayscale images. This corresponds to change the optimization scheme as (5.1). The time step value can be here taken as $\sigma = 1/\left(2 + \max\left(\max(||\nabla u^{\ell+1}||), \max(||\nabla v^{\ell+1}||)\right)\right)$ and $\tau = 1/4$. With parameter $\lambda = 1/2$, the following result, containing sharper motion transitions, should be obtained.

这范数到底怎么求？

flowToColor(u, v)

$|\tilde{I}_2 - I_1|$

Notice that the flowToColor function normalizes the vectors with respect to the highest motion vector. This explains the variations of color intensities in the produced images.

If you have time. You can test your algorithms and compare your results with ground truth motion fields using data available here.

References

[1] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011.

[2] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[3] C. Militello, L. Rundo, and M. C. Gilardi. Applications of imaging processing to mrgfus treatment for fibroids: a review. *Translational Cancer Research*, 3(5), 2014.

[4] L. Rakêt, L. Roholm, M. Nielsen, and F. Lauze. Tv-l1 optical flow for vector valued images. In *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR'11)*, pages 329–343, 2011.

[5] K. Wong, R. Kelso, S. Worthley, P. Sanders, J. Mazumdar, and D. Abbott. Theory and validation of magnetic resonance fluid motion estimation using intensity flow data. *PLoS ONE*, 4(3):1–15, 2009.

[6] C. Zach, T. Pock, and H. Bischof. A globally optimal algorithm for robust tv-l1 range image integration. In *IEEE International Conference on Computer Vision (ICCV'07)*, 2007.