

# Joint Super-Resolution and Optical Flow Estimation

## 1 Energy Functional

Let  $\Omega_l \subset \mathbb{R}$  be a discretized rectangular domain with  $w \times h$  pixels and  $\Omega_h \subset \mathbb{R}$  a domain with  $W \times H$  pixels. Let  $f_1, \dots, f_n : \Omega_l \rightarrow \mathbb{R}^k$  be low resolution color input images with  $k$  color channels. We seek to jointly estimate high resolution images  $u_1, \dots, u_n : \Omega_h \rightarrow \mathbb{R}^k$  and optical flow fields  $v_1, \dots, v_{n-1} : \Omega_h \rightarrow \mathbb{R}^2$  from the low resolution images.

We model the problem in terms of energy minimization of the following functional:

$$E(u, v) = \sum_{i=1}^n \alpha \|Au_i - f_i\|_1 + \beta TV(u_i) + \gamma \sum_{i=1}^{n-1} \sum_{x \in \Omega_h} \|u_i(x) - u_{i+1}(x + v_i(x))\|_1 + TV(v_i^1) + TV(v_i^2) \quad (1)$$

Here  $A : (\Omega_h \rightarrow \mathbb{R}^k) \rightarrow (\Omega_l \rightarrow \mathbb{R}^k)$  is a linear operator which maps a high-resolution image to a low resolution image by blurring it with a gaussian kernel and downsampling it.

$TV(u) := \sum_{x \in \Omega} \|\nabla u(x)\|_2$  denotes the TV regularizer.

## 2 Optimization

Since the energy is hard to minimize jointly in  $u$  and  $v$  we employ a block-coordinate descent approach:

$$\begin{aligned} v^{k+1} &= \arg \min_v E(u^k, v), \\ u^{k+1} &= \arg \min_u E(u, v^{k+1}). \end{aligned} \quad (2)$$

### 2.1 Solving the Problem in $v$ (TV-L1 Optical Flow).

For fixed  $u^k$  the problem reads:

$$v^{k+1} = \arg \min_v \gamma \sum_{x \in \Omega_h} \sum_{i=1}^{n-1} \|u_i^k(x) - u_{i+1}^k(x + v_i(x))\|_1 + TV(v_i^1) + TV(v_i^2) \quad (3)$$

For simplicity, we first consider the case  $n = 2$ :

$$v^{k+1} = \arg \min_v \gamma \sum_{x \in \Omega_h} \|u_1^k(x) - u_2^k(x + v_1(x))\|_1 + TV(v_1^1) + TV(v_1^2) \quad (4)$$

This energy is nonconvex in  $v$ , due to the first term. Thus we linearize it using the first order Taylor expansion,

$$\|u_i^k(x) - u_{i+1}^k(x + v_i(x))\|_1 \approx \|u_1^k(x) - u_2^k(x) - \nabla u_2^k(x)^T v_i(x)\|_1,$$

and end up at the following convex problem:

$$v^{k+1} = \arg \min_v \gamma \sum_{x \in \Omega_h} \underbrace{\|u_1^k(x) - u_2^k(x)\|_1}_{=: -b(x)} - \underbrace{\nabla u_2^k(x)^T v(x)}_{=: (Av)(x)} + TV(v^1) + TV(v^2), \quad (5)$$

where  $v = (v^1, v^2)$ .

### 2.1.1 Primal-Dual Optimization

Since the energy is non-differentiable, a gradient descent based approach does not work. We employ the primal-dual algorithm described in [1, 2] to minimize the energy. First, we rewrite (5) as an equivalent saddle-point problem:

$$\min_v \max_{p \in C, q_1 \in D, q_2 \in D} \langle p, Av + b \rangle + \langle q_1, \nabla v^1 \rangle + \langle q_2, \nabla v^2 \rangle \quad (6)$$

The update equations for the algorithm then read:

$$\begin{aligned} p^{k+1}(x) &= \text{proj}_C(p^k(x) + \sigma_p(x)((Av^k)(x) + b(x))) \\ q_1^{k+1}(x) &= \text{proj}_D(q_1^k(x) + \sigma_q(\nabla v_1^k)(x)), \\ q_2^{k+1}(x) &= \text{proj}_D(q_2^k(x) + \sigma_q(\nabla v_2^k)(x)), \\ \bar{p}^{k+1}(x) &= 2p^{k+1} - p^k, \\ \bar{q}_1^{k+1}(x) &= 2q_1^{k+1} - q_1^k, \\ \bar{q}_2^{k+1}(x) &= 2q_2^{k+1} - q_2^k, \\ v^{k+1}(x) &= v^k - \tau(x)((A^T \bar{p}^{k+1})(x) - (\text{div } \bar{q}_1^{k+1})(x) - (\text{div } \bar{q}_2^{k+1})(x)). \end{aligned} \quad (7)$$

The sets  $C$  and  $D$  are defined as

$$\begin{aligned} C &= \{x \in \mathbb{R} \mid |x| \leq \gamma\}, \\ D &= \{x \in \mathbb{R}^{2n_c} \mid \|x\|_2 \leq 1\}, \end{aligned} \quad (8)$$

and the projections  $\text{proj}_C$ ,  $\text{proj}_D$  can be implemented as an orthogonal projection on a sphere. Only project if you lie outside of the constraint.

The step sizes are chosen according to the scheme described in [2] (see Lemma 2, equation 10, we set  $\alpha = 1$ ):

$$\begin{aligned} \sigma_p(x) &= \frac{1}{\sum_j |A(x, j)|}, \\ \sigma_q &= \frac{1}{2}, \\ \tau(x) &= \frac{1}{2 + 2 + \sum_i |A(i, x)|}, \end{aligned} \quad (9)$$

where  $A(x, j)$  denotes the element in row  $x$  and column  $j$ .

Allocate memory for the variables  $p \in \mathbb{R}^{w*h*n_c}$ ,  $q_1 \in \mathbb{R}^{w*h*2*n_c}$ ,  $q_2 \in \mathbb{R}^{w*h*2*n_c}$ ,  $\bar{p}, \bar{q}_1, \bar{q}_2, v \in \mathbb{R}^{w*h*2*n_c}$  as `float` arrays and implement CUDA kernels for the update equations of the primal-dual algorithm. One kernel should perform the update in  $p, q_1, q_2$  and do the overrelaxation, the other kernel should do the update in  $v$ .

## 2.2 Solving the Problem in $u$ .

$$\begin{aligned}
u_i^{k+1} &= u_i^k - \tau_i (A^T \bar{p}_i^{k+1} - \text{div}(\bar{q}_i^{k+1}) + B^T \bar{r}^{k+1}), \\
q_i^{k+1} &= \text{proj}_E(q_i^k + \sigma_q(\nabla u_i)), \\
p_i^{k+1} &= \text{proj}_F(p_i^k + \sigma_p(Au_i^k - f_i)), \\
r^{k+1} &= \text{proj}_G(r^{k+1} + \sigma_r B \begin{pmatrix} u_1^k \\ u_2^k \end{pmatrix}), \\
\bar{q}_i^{k+1} &= 2q_i^{k+1} - q_i^k; \quad \bar{p}_i^{k+1}, \quad \text{and} \quad \bar{r}^{k+1} \text{ analogously}
\end{aligned} \tag{10}$$

$$\begin{aligned}
\sigma_q &= \frac{1}{2}, \\
\sigma_p &= 1, \\
\tau_i &= \frac{1}{1 + 4 + ?(x)}, \\
?(x) &= \begin{cases} 1 & i = 1 \\ 1 + |(v/u?)^1(x)| * 2 + |(v/u?)^2(x)| * 2 & i \geq 2 \end{cases} \\
\sigma_r(x_{(ijc)}) &= \frac{1}{2 + 2|(u/v?)^1(x_{(ij)})| + 2|(u/v?)^2(x)|}
\end{aligned} \tag{11}$$

$$\begin{aligned}
B &= (I, \quad -I - (u/v?)^1 \delta_x - (u/v?)^2 \delta_y), \\
B^T &= \begin{pmatrix} I \\ -I + \underset{\text{dirichlet}^x}{bw\_diff}(u/v?)^1 + \underset{\text{dirichlet}^y}{bw\_diff}(u/v?)^2 \end{pmatrix}, \\
A &= DB, \\
A^T &= BD^T
\end{aligned} \tag{12}$$

$$\begin{aligned}
E &= \{x \in \mathbb{R} \mid \|x\|_2 \leq \beta\}, \\
F &= \{x \in \mathbb{R} \mid |x| \leq \alpha\}, \\
G &= \{x \in \mathbb{R} \mid |x| \leq \gamma\}
\end{aligned} \tag{13}$$

$$\begin{aligned}
\delta_x = D_x &= \begin{pmatrix} 1 & -1 & & \\ & \ddots & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix}, \\
v_x &= \begin{pmatrix} v_x(?, ?) & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \ddots & v_x(?, ?) \end{pmatrix}
\end{aligned} \tag{14}$$

$$\begin{aligned}
A &: \mathbb{R}^{W*H*c} \mapsto \mathbb{R}^{w*h*c}, \\
B_{flow} &: \mathbb{R}^{2*W*H*c} \mapsto \mathbb{R}^{W*H*c}, \\
B_l &: \mathbb{R}^{W*H*c} \mapsto \mathbb{R}^{W*H*c}, \\
D &: \mathbb{R}^{W*H*c} \mapsto \mathbb{R}^{w*h*c}
\end{aligned} \tag{15}$$

## References

- [1] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.*, 40:120–145, 2011.
- [2] T. Pock and A. Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *ICCV*, pages 1762–1769, 2011.