Scene Flow by Tracking in Intensity and Depth Data

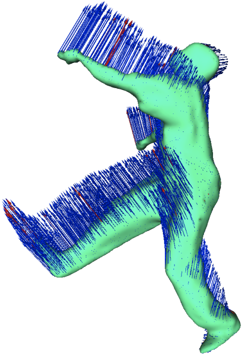
Julian Quiroga Frederic Devernay James Crowley

PRIMA team, **INRIA Grenoble**

julian.quiroga@inria.fr

June 12, 2012

Motivation



Surface Flow, Morpheo-INRIA 2011

Messing et al., ICCV 2009

Scene flow computation

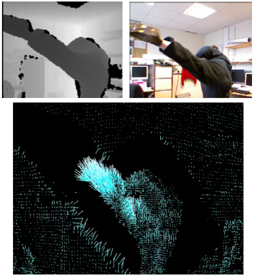
Stereo or multiview:

- From several optical flows
- By using structure constrains and 2D/3D regularization
- Simultaneously with 3D surface
- Tracking surfels (surface elements)

Color and depth:

- Photometric constrains and 3D regularization
- Particle filtering in 3D

Basha et al., CVPR 2010



Hadfield and Bowden, ICCV 2011

Our work

Approach

Sparse scene flow: we track small surface patches in the scene by using a pair of aligned intensity and depth images.

Model

To constraint the scene flow in the image domain we assume a scene composed of rigidly moving 3D parts performing translation.

Framework

By using the scene flow as parameter vector we extend the Lukas-Kanade approach to exploit both intensity and depth data.

Result

We simultaneously solve for the scene flow and the image flow.

Presentation outline

- **Lucas-Kanade framework**
- Motion model
- Locally rigid tracking approach
- Tracking in intensity and depth
- Experimentation
- Conclusion

Lucas-Kanade Framework

The goal of Lucas-Kanade algorithm is to align a template image $T(\mathbf{x})$ to an input image $I(\mathbf{x})$. This problem can be stated as

$$\mathbf{P} = \arg\min_{\mathbf{P}} \sum_{\mathbf{x}} \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{P})\right) - T\left(\mathbf{x}\right)\right]^2$$

Assuming that an initial estimate of **P** is known, each optimization step finds $\Delta\mathbf{P}$ which minimizes

$$\sum_{\mathbf{x}} \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{P}+\Delta\mathbf{P})\right) - T\left(\mathbf{x}\right)\right]^2 = \sum_{\mathbf{x}} \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{P})\right) + \nabla_I \frac{\partial \mathbf{W}}{\partial \mathbf{P}} \Delta\mathbf{P} - T\left(\mathbf{x}\right)\right]^2$$

Taking the partial derivative with respect to $\Delta\mathbf{P}$ and solving it gives

$$\boxed{\Delta\mathbf{P} = \mathbf{H}^{-1} \sum_{\mathbf{x}} \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{P}}\right)^T \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{P})\right) - T\left(\mathbf{x}\right)\right]}$$

where $\mathbf{H} = \sum_{\mathbf{x}} \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{P}}\right)^T \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{P}}\right)$.

Presentation outline

- Lucas-Kanade framework
- **Motion model**
- Locally rigid tracking approach
- Tracking in intensity and depth
- Experimentation
- Conclusion

Motion model

The instantaneous motion of a rigid surface point can be expressed as

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{V},$$

The 3D motion of the surface generates the image flow given by

$$u = x' - x = \left(\frac{X - Y\Omega_Z + Z\Omega_Y + V_X}{Z - X\Omega_Y + Y\Omega_X + V_Z} - \frac{X}{Z} \right) f_x$$

and

$$v = y' - y = \left(\frac{Y - X\Omega_Z + Z\Omega_X + V_y}{Z - X\Omega_Y + Y\Omega_X + V_Z} - \frac{Y}{Z} \right) f_y$$

Assuming that the inter-frame rotation is negligible , the image flow induced on a pixel $\mathbf{x} = (x, y)$ by the 3D translation of the surface can be modeled as follows

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} 1 & 0 & -x \\ 0 & 1 & -y \end{pmatrix} \begin{pmatrix} V_X \\ V_Y \\ V_Z \end{pmatrix} = \Delta(\mathbf{x}; \mathbf{V})$$

Presentation outline

- Lucas-Kanade framework
- Motion model
- **Locally rigid tracking approach**
- Tracking in intensity and depth
- Experimentation
- Conclusion

Locally rigid tracking approach

Under brightness constancy assumption, points $\mathbf{X}$ at time $t-1$ and $\mathbf{X}' = \mathbf{X} + \mathbf{V}$ at time t are projected with the same intensity in the image

$$I^t\left(\hat{\mathbf{M}}\left(\mathbf{X}+\mathbf{V}\right)\right) = I^{t-1}\left(\hat{\mathbf{M}}(\mathbf{X})\right)$$

Considering a set of surface points $\mathbf{S}$, the scene flow computation is stated as finding vector $\mathbf{V} = \{V_X, V_Y, V_Z\}$ which minimizes

$$\sum_{\mathbf{X}\in\mathbf{S}}\left[I^t\left(\hat{\mathbf{M}}\left(\mathbf{X}+\mathbf{V}\right)\right) - I^{t-1}\left(\hat{\mathbf{M}}\left(\mathbf{X}\right)\right)\right]^2$$

The imagen flow of each surface points is given by the warp function

$$\mathbf{W}(\mathbf{x};\mathbf{V}) = \hat{\mathbf{M}}\left(\mathbf{X}+\mathbf{V}\right) = \mathbf{x} + \Delta(\mathbf{x};\mathbf{V})$$

where $\Delta(\mathbf{x};\mathbf{V})$ is the proposed motion model.

Locally rigid tracking approach

The problem can be formulated in the image domain as follows

$$\mathbf{V} = \arg\min_{\mathbf{V}} \sum_{\{\mathbf{x}\}\in\mathbf{s}} \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{V})\right) - T\left(\mathbf{x}\right)\right]^2$$

Solution. Each element of the Jacobian is given by

$$\frac{\partial \mathbf{W}}{\partial \mathbf{V}} = \frac{1}{Z(\mathbf{x})} \begin{pmatrix} 1 & 0 & -x \\ 0 & 1 & -y \end{pmatrix}$$

Over each iteration the Hessian matrix can be expressed as

$$\mathbf{H} = \sum_{\{\mathbf{x}\}} \frac{1}{Z(\mathbf{x})^2} \begin{pmatrix} I_x^2 & I_x I_y & I_x I_\Sigma \\ I_x I_y & I_y^2 & I_y I_\Sigma \\ I_x I_\Sigma & I_y I_\Sigma & I_\Sigma^2 \end{pmatrix}$$

with $I_\Sigma = -\left(x I_x + y I_y\right)$.

Presentation outline

- Lucas-Kanade framework
- Motion model
- Locally rigid tracking approach
- **Tracking in intensity and depth**
- Experimentation
- Conclusion

Tracking in intensity and depth

Under translation **V** the depth image must satisfy

$$Z^t(\hat{\mathbf{M}}(\mathbf{X}+\mathbf{V})) = Z^{t-1}(\hat{\mathbf{M}}(\mathbf{X})) + V_Z$$

Therefore, we propose to formulate the scene flow computation by constraining **V** both in the intensity and depth:

$$\sum_{\{\mathbf{x}\}\in\mathbf{s}} \left[I\left(\mathbf{W}(\mathbf{x};\mathbf{V})\right) - T\left(\mathbf{x}\right)\right]^2 + \lambda\left[Z\left(\mathbf{W}(\mathbf{x};\mathbf{V})\right) - \left(T_z\left(\mathbf{x}\right) + D^T\mathbf{V}\right)\right]^2$$

where $\lambda = \sigma_I^2/\sigma_Z^2$ and $D = (0,0,1)$ separates the Z component.
Solution.

$$\mathbf{H} = \sum_{\{\mathbf{x}\}} \frac{1}{Z(\mathbf{x})^2} \begin{pmatrix} I_x^2 + \lambda Z_x^2 & I_x I_y + \lambda Z_x Z_y & I_x I_\Sigma + \lambda Z_x\left(Z_\Sigma - 1\right) \\ I_x I_y + \lambda Z_x Z_y & I_y^2 + \lambda Z_y^2 & I_y I_\Sigma + \lambda Z_y\left(Z_\Sigma - 1\right) \\ I_x I_\Sigma + \lambda Z_x\left(Z_\Sigma - 1\right) & I_y I_\Sigma + \lambda Z_y\left(Z_\Sigma - 1\right) & I_\Sigma^2 + \lambda(Z_\Sigma - 1)^2 \end{pmatrix}$$

with $I_\Sigma = -\left(xI_x + yI_y\right)$ and $Z_\Sigma = -\left(xZ_x + yZ_y\right)$.

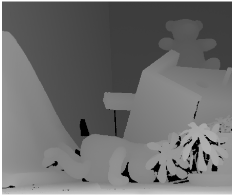
Presentation outline

- Lucas-Kanade framework
- Motion model
- Locally rigid tracking approach
- Tracking in intensity and depth
- **Experimentation**
- Conclusion

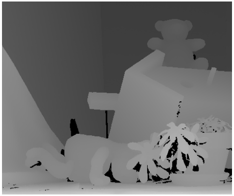
Experimentation - Middlebury datastes

I_1

I_2



Z_1



Z_2

Details

- Images : Teddy, Cones (2 and 6)
- 5 leves of PYR decomposition
- Window size: 11×11
- Image coverage: 85%

Error measures

- Optical flow: $\mathrm{RMS}_{\mathrm{OF}}$, $\mathrm{AEE}_{\mathrm{OF}}$, R$X$
- Scene flow: $\mathrm{NRMS}_{\mathrm{V}}$, R$X$%

Comparisons

- RT: proposed method (*rigid traslation*)
- RT0: proposed method with $\lambda = 0$
- KLT: KLT by Bouguet (OpenCV)
- OF_{R}: KLT with a robust norm
- Hug_{07}: Huguet and Devernay, ICCV 2007
- Bas_{10}: Basha et al., CVPR 2010
- Had_{11}: Hadfield and Bowden, ICCV 2011

Experimentation - Middlebury datastes

	RT	RT0	OF_R	KLT
RMS_{OF}	2.51	2.61	4.69	5.95
R1.0	12.9	14.8	28.4	40.1
R5.0	2.32	3.99	15.2	19.7
AEE_{OF}	1.15	1.33	1.56	1.45

Table 1: Errors in the optical flow.

	RT	RT0	OF_R	KLT
$NRMS_V$	11.1	55.9	68.4	82.0
R5%	17.1	28.8	37.9	38.1
R20%	4.97	9.68	17.5	19.1

Table 2: Errors in the scene flow.

	RT			OF_R		
	Tex	*Utex*	*DD*	*Tex*	*Utex*	*DD*
RMS_{OF}	4.99	3.11	6.91	5.75	7.20	7.05
R1.0	16.5	39.8	32.5	38.9	58.8	68.7
$NRMS_V$	23.2	10.9	26.5	96.7	202	188
R5%	12.1	28.5	25.1	32.0	51.5	69.6

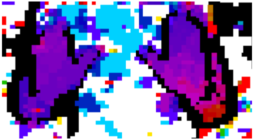
Table 3: Errors by regions.

	RT	Hug_{07}	Bas_{10}	Had_{11}
$RMS_{OF}(\%)$	5.70	6.00	2.96	0.10
AEE_{OF}	2.47	0.60	0.70	5.03

Table 4: Scene flow comparison.

Experimentation - Kinect images

I_1

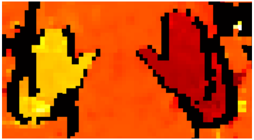


Optical flow

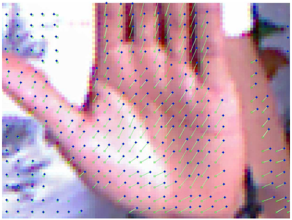
I_2

$\mathbf{V}$

Z_1



V_z



Scene flow projection

Presentation outline

- Lucas-Kanade framework
- Motion model
- Locally rigid tracking approach
- Tracking in intensity and depth
- Experimentation
- **Conclusion**

Conclusion

- We have proposed a method to compute a sparse scene flow by using an aligned pair of intensity and depth images.
- Modeling the image flow as a function of the 3D motion field with help from the depth sensor allows the constraint of the scene flow of a small surface patch in the image domain.
- Combining intensity and depth data in a Lucas-Kanade framework we simultaneoulsy solve for the scene flow and image flow.
- This method is versatile and cand be used to generate more accurate trajectories or to define scene flow based descriptors.

Future work

- A criterion for selecting good regions to track
- Experimentation: action and gesture recognition

The End