ECCV '14
Zürich, September 6-12th, 2014
European Conference on Computer Vision

Dense Semi-Rigid Scene Flow Estimation from RGBD images

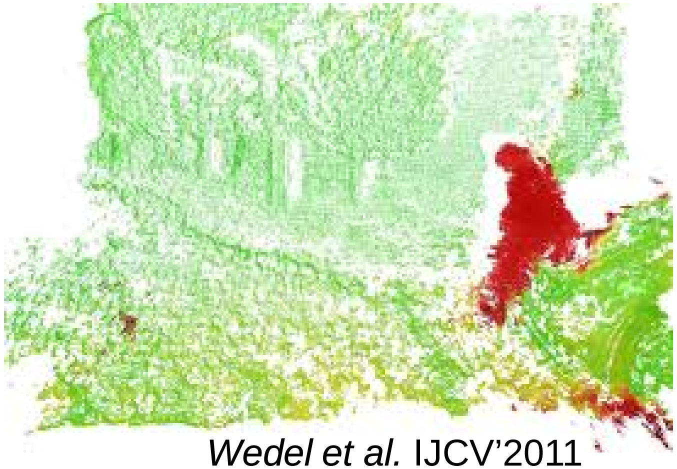
Julián Quiroga, Thomas Brox, Frédéric Devernay, James Crowley




Inria
INVENTEURS DU MONDE NUMÉRIQUE
UNI FREIBURG
Pontificia Universidad JAVERIANA

Motivation

The **scene flow** is the 3D motion field of the scene (*Vedula et al.* ICCV'99).

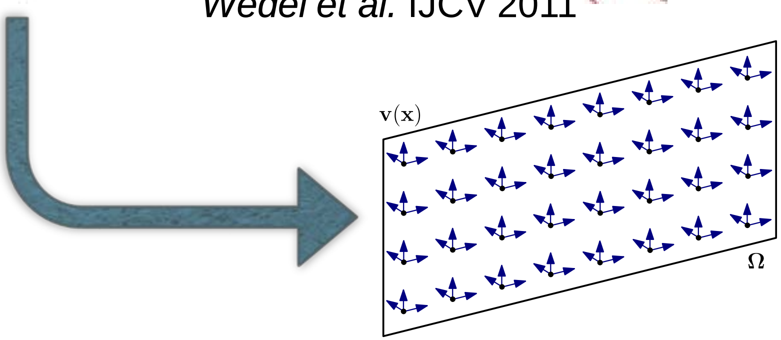


Wedel et al. IJCV'2011

Motivation

The **scene flow** is the 3D motion field of the scene (*Vedula et al.* ICCV'99).

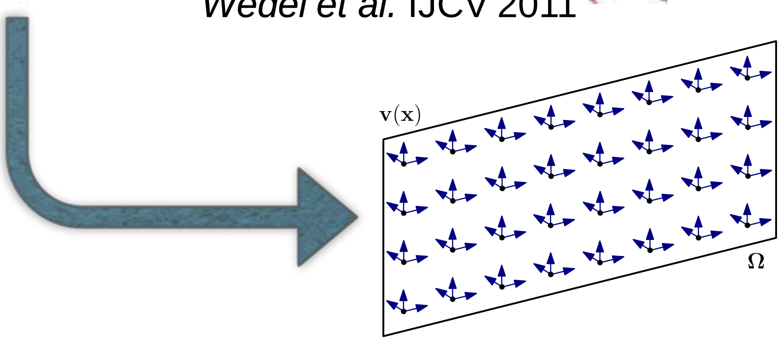
Wedel et al. IJCV'2011



Motivation

The **scene flow** is the 3D motion field of the scene (*Vedula et al.* ICCV'99).

Wedel et al. IJCV'2011



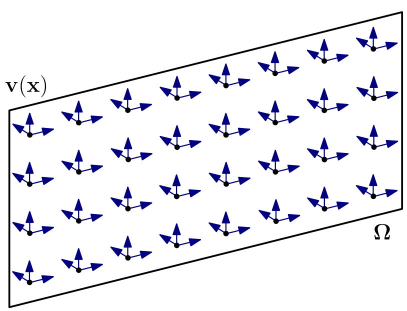
Applications:

- 3D motion capture
- 3D modeling on non-rigid objects
- Autonomous navigation
- Interaction
- …

Motivation

The **scene flow** is the 3D motion field of the scene (*Vedula et al.* ICCV'99).

Wedel et al. IJCV'2011



Applications:

- 3D motion capture
- 3D modeling on non-rigid objects
- Autonomous navigation
- Interaction
- …

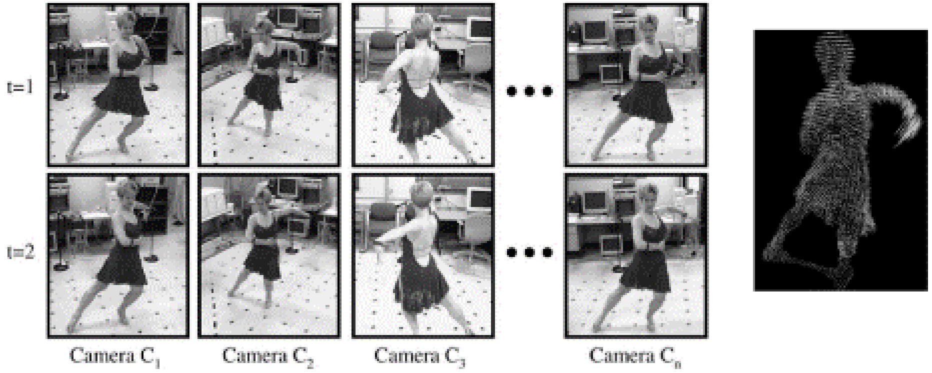
Estimation:

- Unknown 3D structure
- Given the 3D structure

Scene Flow Estimation

from stereo or multi-view

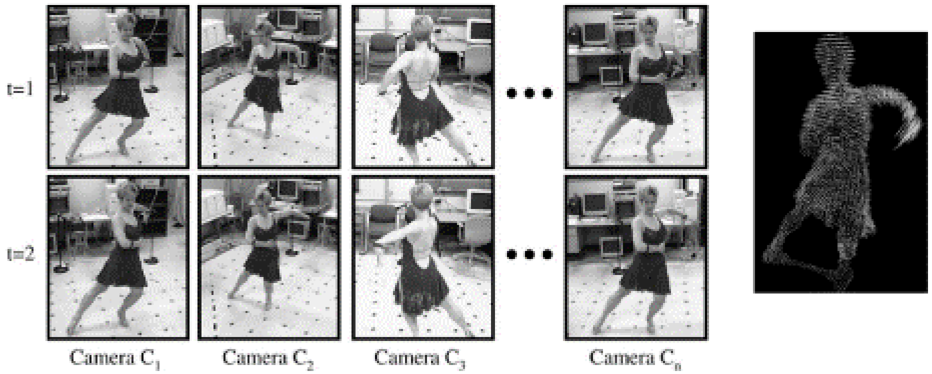
✪ Computing several optical flows (*Vedula et al.* PAMI'05):



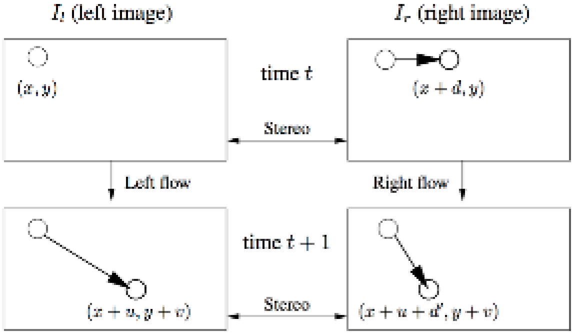
Scene Flow Estimation

from stereo or multi-view

✪ Computing several optical flows (*Vedula et al.* PAMI'05):



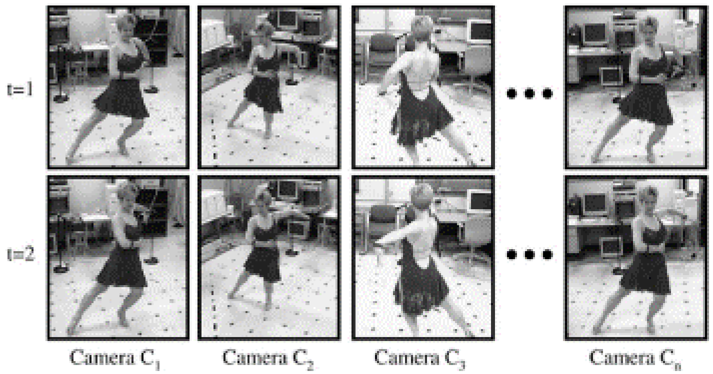
✪ Using constraints from the camera system: (*Huguet & Devernay* ICC V'07, *Wedel et al.* ECCV'08, *Basha et al.* CVPR'10)

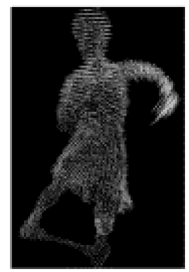


Scene Flow Estimation

from stereo or multi-view

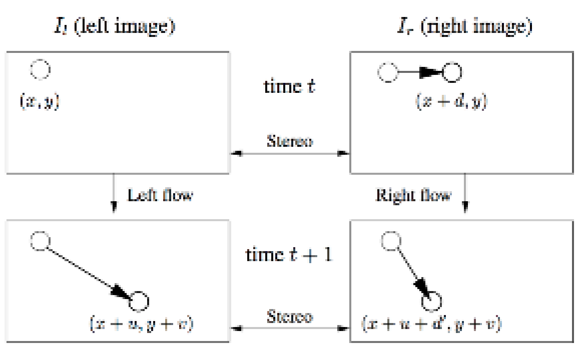
- ✪ Computing several optical flows (*Vedula et al.* PAMI'05):





- ✪ Encouraging locally or piecewise rigid motions (*Vogel et al.* ICCV'11 '13):

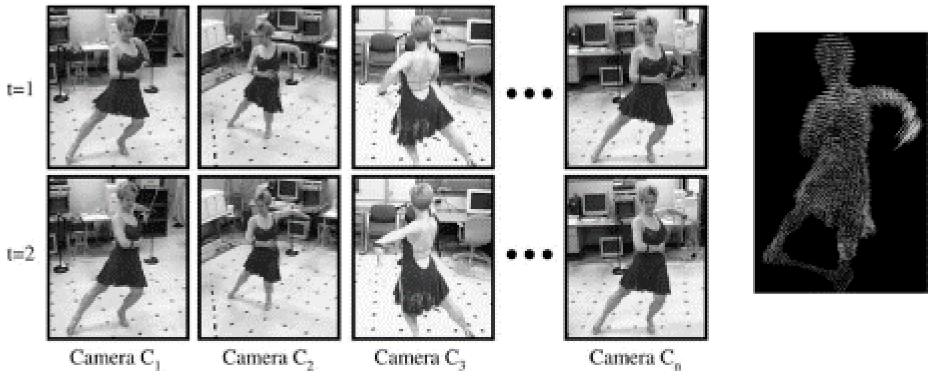
- ✪ Using constraints from the camera system: (*Huguet & Devernay* ICC V'07, *Wedel et al.* ECCV'08, *Basha et al.* CVPR'10)



Scene Flow Estimation

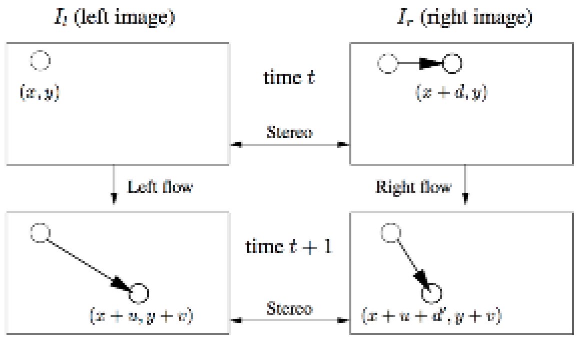
from stereo or multi-view

✪ Computing several optical flows (*Vedula et al.* PAMI'05):



✪ Encouraging locally or piecewise rigid motions (*Vogel et al.* ICCV'11 '13):

✪ Using constraints from the camera system: (*Huguet & Devernay* ICC V'07, *Wedel et al.* ECCV'08, *Basha et al.* CVPR'10)

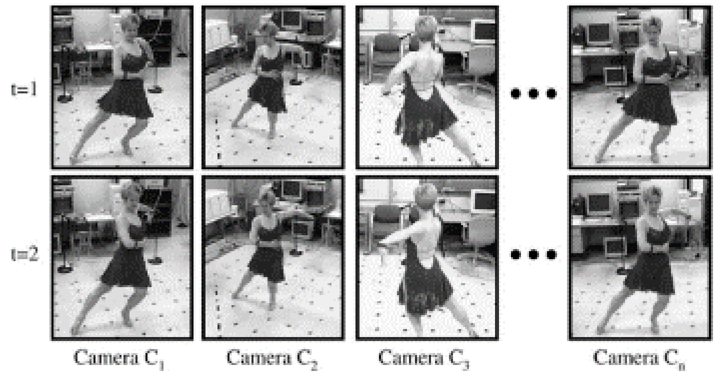


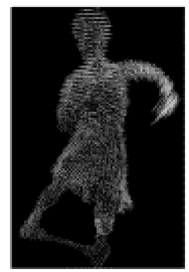
If a depth sensor is available?

Scene Flow Estimation

from stereo or multi-view

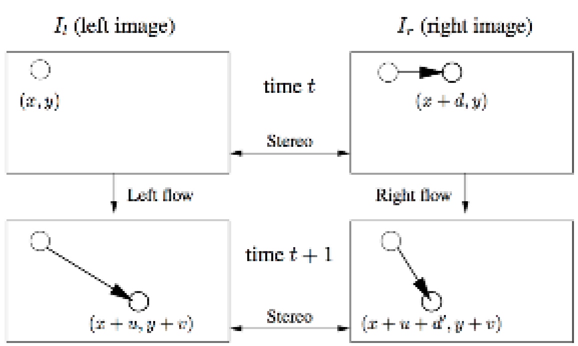
✪ Computing several optical flows (*Vedula et al.* PAMI'05):





✪ Encouraging locally or piecewise rigid motions (*Vogel et al.* ICCV'11 '13):

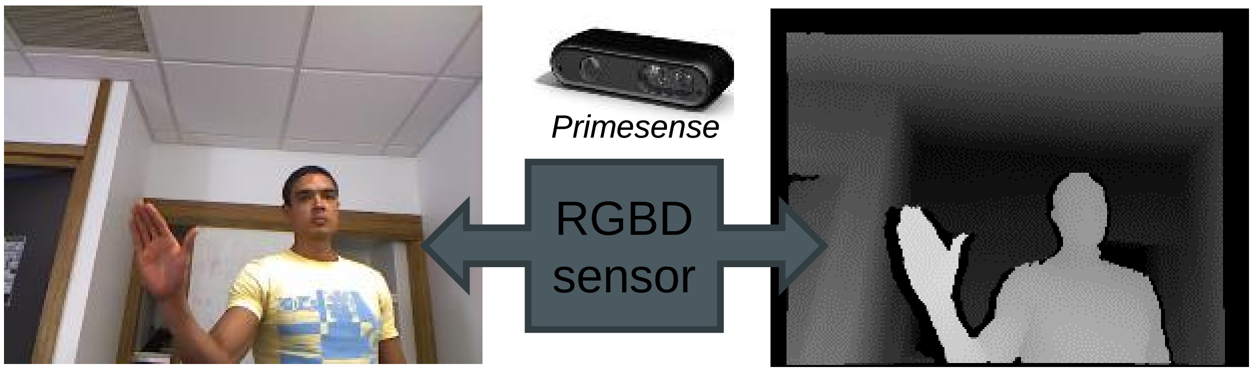
✪ Using constraints from the camera system: (*Huguet & Devernay* ICC V'07, *Wedel et al.* ECCV'08, *Basha et al.* CVPR'10)



If a **depth sensor** is available?

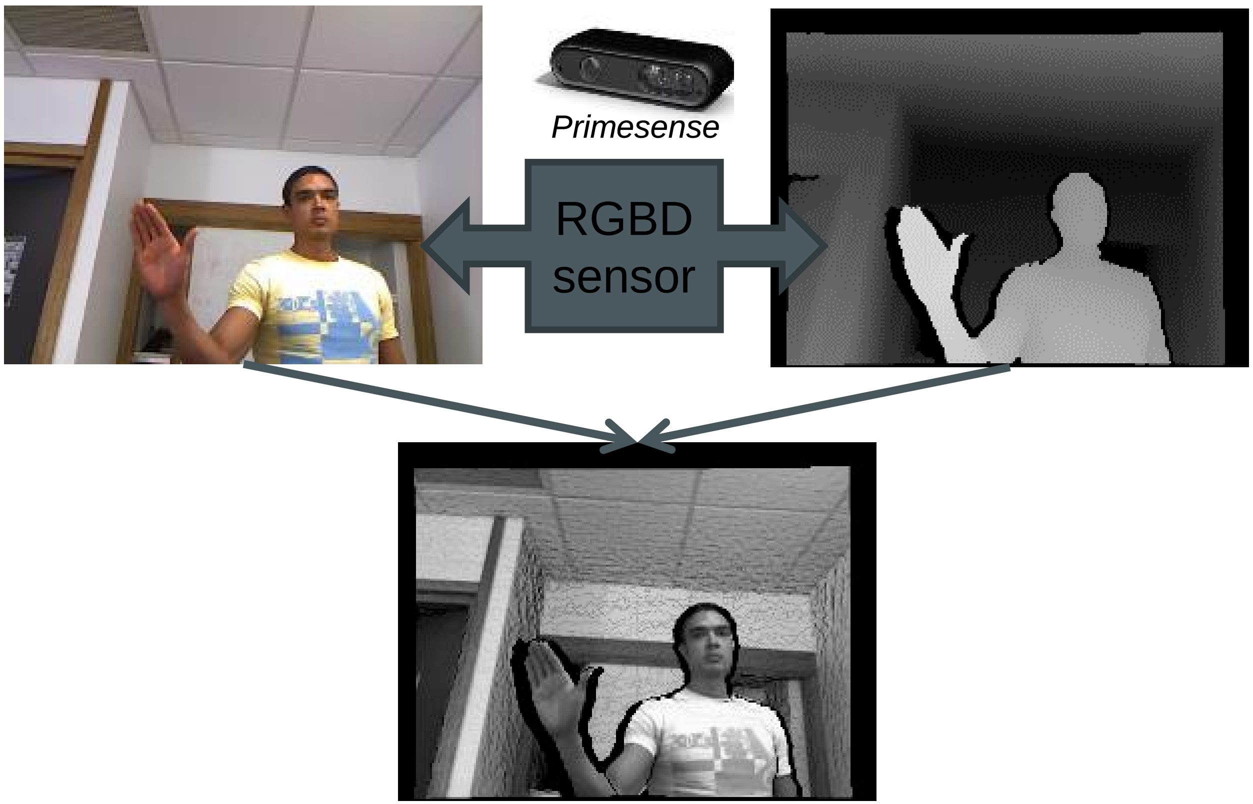
RGBD data

✪ ***RGBD image:*** a registered pair of color and depth images



RGBD data

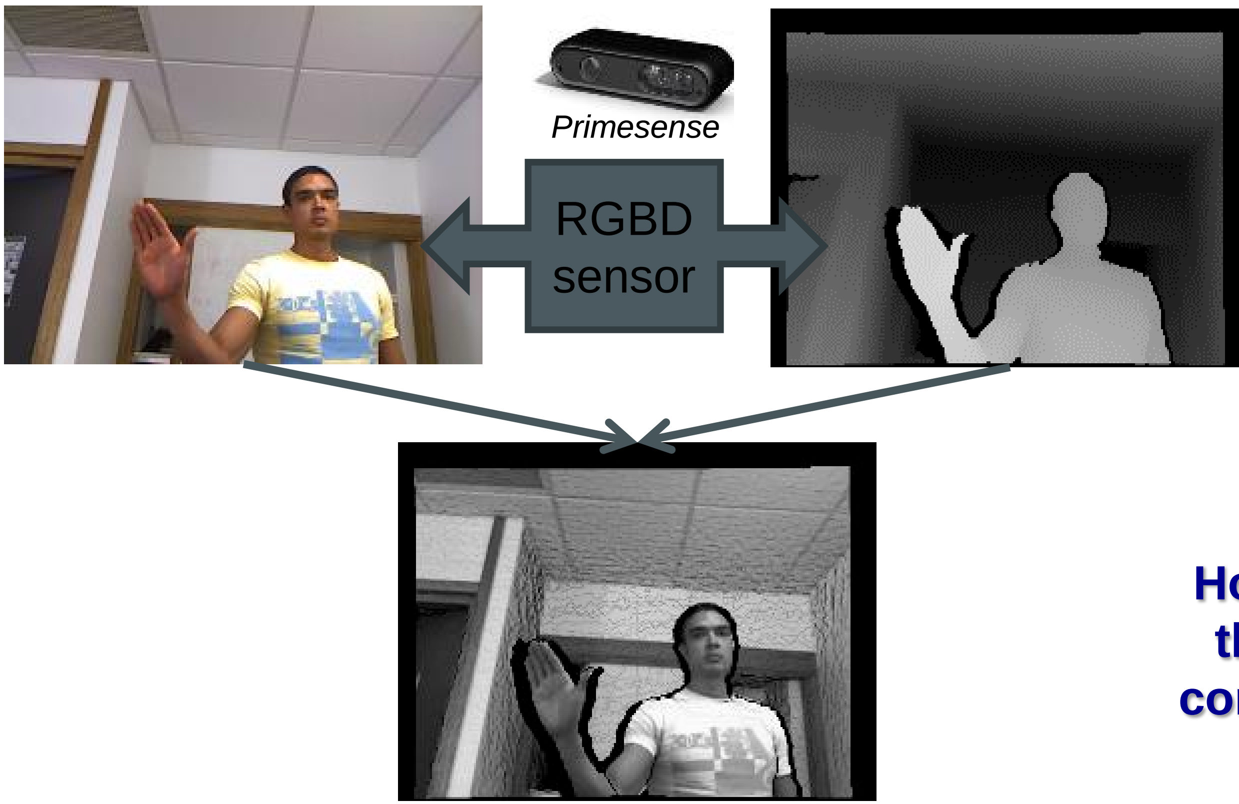
✪ ***RGBD image:*** a registered pair of color and depth images



RGBD representation of the scene

RGBD data

✪ ***RGBD image:*** a registered pair of color and depth images



RGBD representation of the scene

How to fully exploit the RGBD data to compute a confident scene flow?

Scene Flow Estimation

from RGBD images

- Using RGB data as an additional channel (*Spies et al.* CVIU'02, *Lukins et al.* BMVC'04):

$$0 = I_x u + I_y v + I_t$$

Optical flow equation

$$w = Z_x u + Z_y v + Z_t$$

Range flow equation

Scene Flow Estimation

from RGBD images

- Using RGB data as an additional channel (*Spies et al.* CVIU'02, *Lukins et al.* BMVC'04):

$$0 = I_x u + I_y v + I_t$$

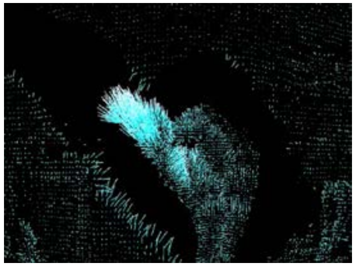
Optical flow equation

$$w = Z_x u + Z_y v + Z_t$$

Range flow equation

- 3D particle filtering

(*Hadfield & Bowden* ICCV'11)



Scene Flow Estimation

from RGBD images

- Using RGB data as an additional channel (*Spies et al.* CVIU'02, *Lukins et al.* BMVC'04):

$$0 = I_x u + I_y v + I_t$$

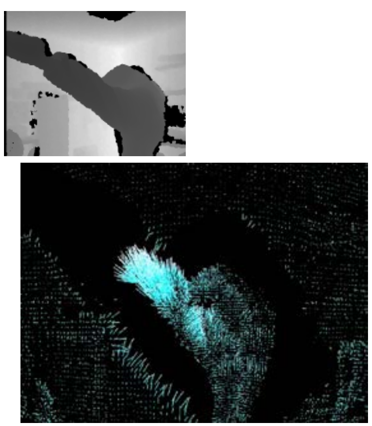
Optical flow equation

$$w = Z_x u + Z_y v + Z_t$$

Range flow equation

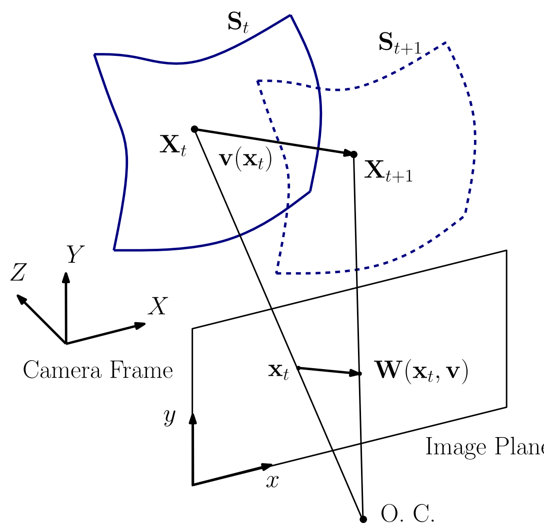
- 3D particle filtering

(*Hadfield & Bowden* ICCV'11)



- Projective warping function

(*Quiroga et al*. CVPRW'12 ,ICIP'13)



Scene Flow Estimation

from RGBD images

- Using RGB data as an additional channel (*Spies et al.* CVIU'02, *Lukins et al.* BMVC'04):

$$0 = I_x u + I_y v + I_t$$

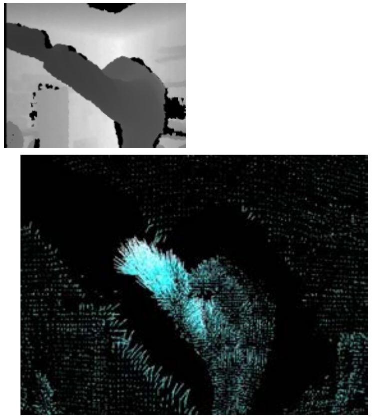
Optical flow equation

$$w = Z_x u + Z_y v + Z_t$$

Range flow equation

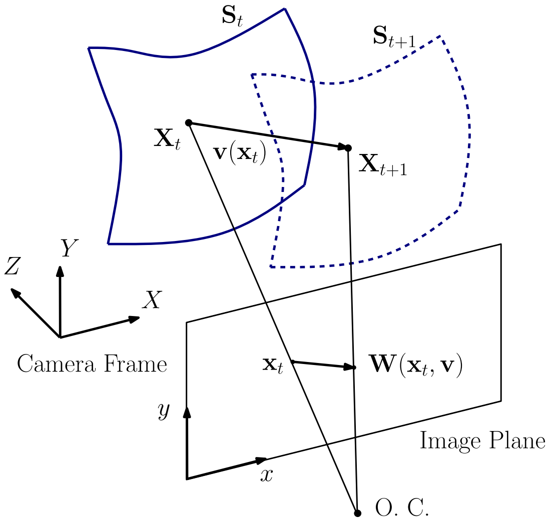
- 3D particle filtering

(*Hadfield & Bowden* ICCV'11)



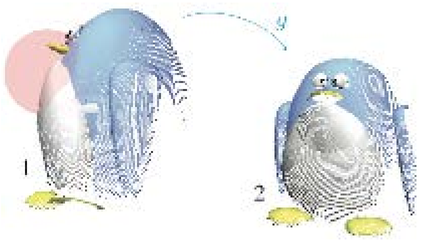
- Projective warping function

(*Quiroga et al*. CVPRW'12 ,ICIP'13)



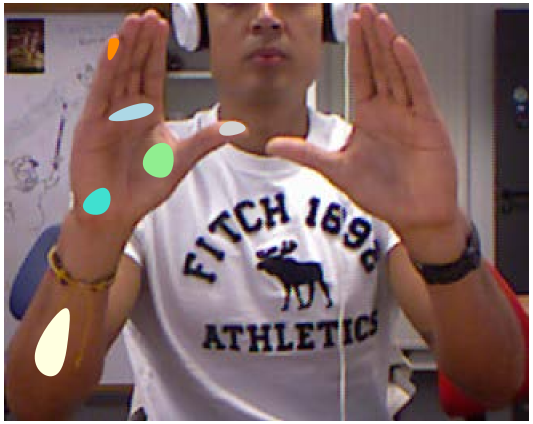
- Matching 3D patches

(*Hornacek et al. CVPR*'14)



Proposed approach

- ***Idea:*** Exploit the semi-rigidity of real-world scenes

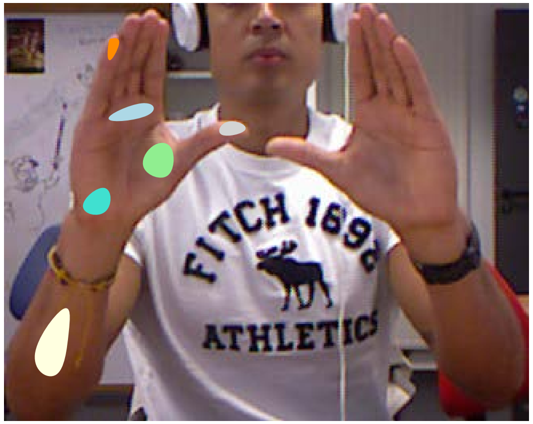


Local rigidity

Piecewise rigidity

Proposed approach

✪ ***Idea:*** Exploit the semi-rigidity of real-world scenes



Local rigidity

Piecewise rigidity

How?

Proposed approach (cont.)

- Using an over-parametrization of the 3D motion

$$\xi(\mathbf{x}) : \mathbf{\Omega} \rightarrow \mathbb{R}^6$$

Motion field of rigid motions

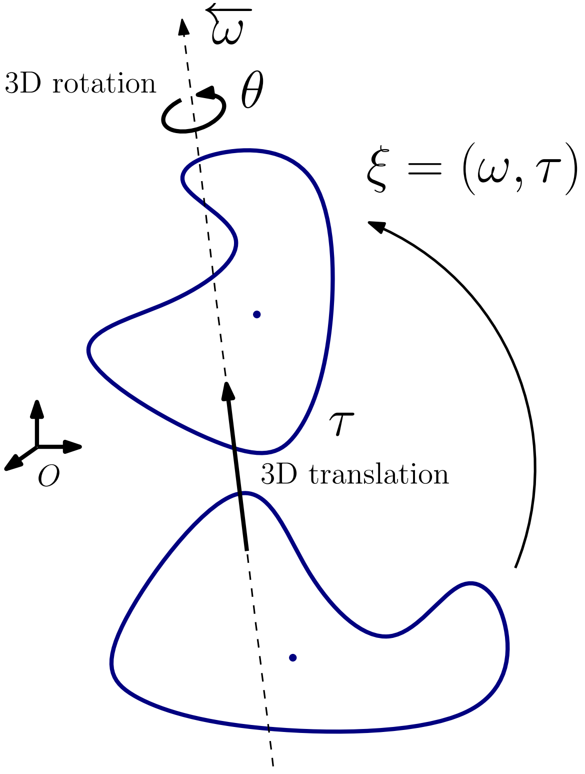
Proposed approach (cont.)

- Using an over-parametrization of the 3D motion

$$\xi(\mathbf{x}) : \mathbf{\Omega} \rightarrow \mathbb{R}^6$$

Motion field of rigid motions

*Every scene point follows a **twist motion***



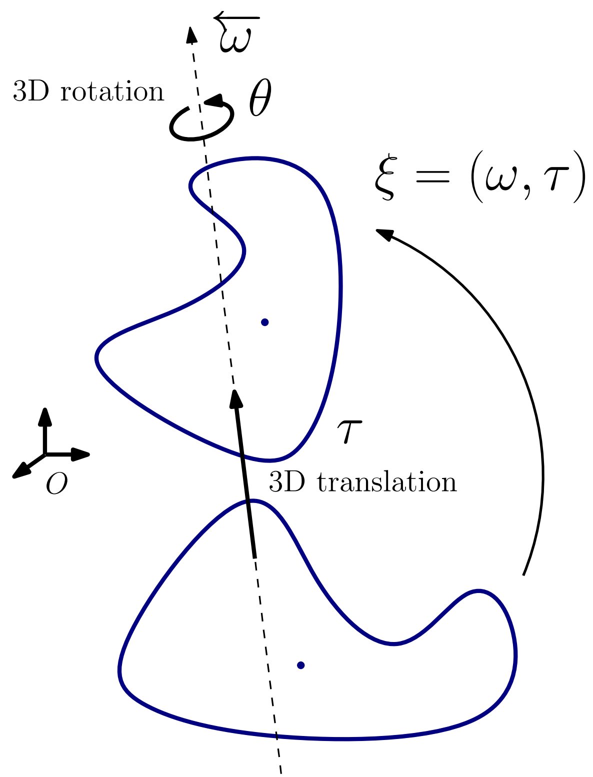
Proposed approach (cont.)

- Using an over-parametrization of the 3D motion

$$\xi(\mathbf{x}) : \mathbf{\Omega} \rightarrow \mathbb{R}^6$$

Motion field of rigid motions

*Every scene point follows a **twist motion***



- Scene flow recovery: $\boxed{\tilde{\mathbf{X}}_{t+1} = e^{\hat{\xi}(\mathbf{x}_t)}\tilde{\mathbf{X}}_t}$

Exponential map

Proposed approach (cont.)

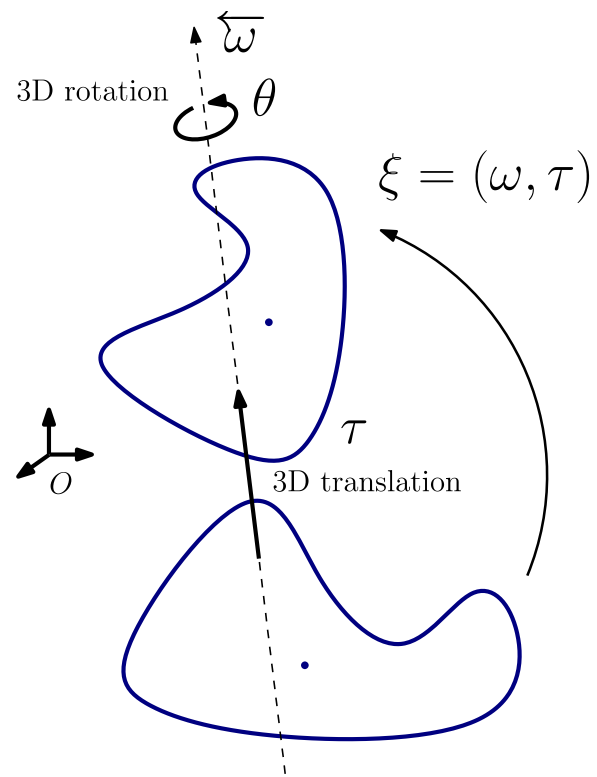
- Using an over-parametrization of the 3D motion

$$\xi(\mathbf{x}) : \mathbf{\Omega} \rightarrow \mathbb{R}^6$$

Motion field of rigid motions

⬇

*Every scene point follows a **twist motion***



- Scene flow recovery: $\boxed{\tilde{\mathbf{X}}_{t+1} = e^{\hat{\xi}(\mathbf{x}_t)} \tilde{\mathbf{X}}_t}$

Exponential map

- We focus on two-frames scene flow estimation

Semi-rigid Scene Flow Energy

$$E(\xi) = E_D(\xi) + \alpha E_S(\xi)$$

Semi-rigid Scene Flow Energy

$$E(\xi) = \boxed{E_D(\xi)} + \alpha E_S(\xi)$$

Data term

- Maximizes consistency between the motion field and the RGBD images
- Encourages **locally-rigid motions**

Semi-rigid Scene Flow Energy

$$E(\xi) = \boxed{E_D(\xi)} + \boxed{\alpha E_S(\xi)}$$

Data term

- Maximizes consistency between the motion field and the RGBD images
- Encourages **locally-rigid motions**

Smoothness term

- Favors **piecewise rigid solutions**
- Preserves motion discontinuities

Semi-rigid Scene Flow Energy

$$E(\xi) = \boxed{E_D(\xi)} + \boxed{\alpha E_S(\xi)}$$

Data term

- Maximizes consistency between the motion field and the RGBD images
- Encourages **locally-rigid motions**

Smoothness term

- Favors **piecewise rigid solutions**
- Preserves motion discontinuities

An adjustable combination of both!

Data term

- **Consistency between motion and RGBD data**

$$\mathbf{x}_{t+1} = \mathbf{W}(\mathbf{x}_t, \xi)$$

Warping function

Data term

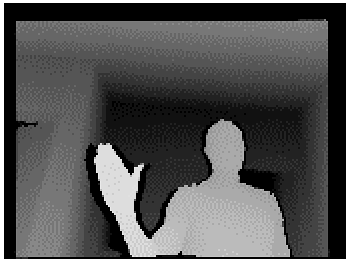
- **Consistency between motion and RGBD data**

$$\mathbf{x}_{t+1} = \mathbf{W}(\mathbf{x}_t, \xi)$$

Warping function

Brightness (I)

Brightness gradient (I_g)



Depth (Z)

Data term

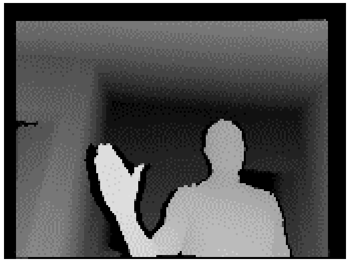
- **Consistency between motion and RGBD data**

$$\mathbf{x}_{t+1} = \mathbf{W}(\mathbf{x}_t, \xi)$$

Warping function

Brightness (I)

Brightness gradient (I_g)



Depth (Z)

$$\rho_{data}(\mathbf{x}, \xi) = \Psi\left(\rho_I^2(\mathbf{x}, \xi) + \gamma\rho_g^2(\mathbf{x}, \xi)\right) + \lambda\Psi\left(\rho_Z^2(\mathbf{x}, \xi)\right)$$

Data term

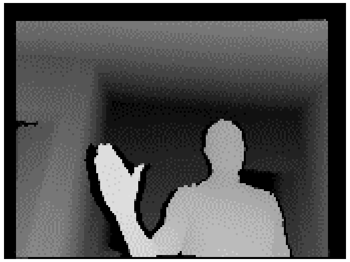
- **Consistency between motion and RGBD data**

$$\mathbf{x}_{t+1} = \mathbf{W}(\mathbf{x}_t, \xi)$$

Warping function

Brightness (I)

Brightness gradient (I_g)

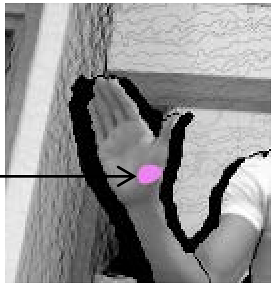


Depth (Z)

$$\rho_{data}(\mathbf{x}, \xi) = \Psi\left(\rho_I^2(\mathbf{x}, \xi) + \gamma \rho_g^2(\mathbf{x}, \xi)\right) + \lambda \Psi\left(\rho_Z^2(\mathbf{x}, \xi)\right)$$

- **Local rigidity**

$$\rho^L(\mathbf{x}, \xi) = \sum_{\mathbf{x}' \in N(\mathbf{x})} \rho_{data}(\mathbf{x}', \xi)$$



Data term

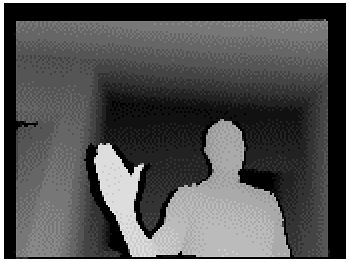
- **Consistency between motion and RGBD data**

$$\mathbf{x}_{t+1} = \mathbf{W}(\mathbf{x}_t, \xi)$$

Warping function

Brightness (I)

Brightness gradient (I_g)

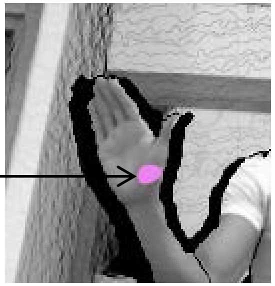


Depth (Z)

$$\rho_{data}(\mathbf{x}, \xi) = \Psi\left(\rho_I^2(\mathbf{x}, \xi) + \gamma \rho_g^2(\mathbf{x}, \xi)\right) + \lambda \Psi\left(\rho_Z^2(\mathbf{x}, \xi)\right)$$

- **Local rigidity**

$$\rho^L(\mathbf{x}, \xi) = \sum_{\mathbf{x}' \in N(\mathbf{x})} \rho_{data}(\mathbf{x}', \xi)$$



$$\boxed{E_D(\xi) = \sum_{\mathbf{x} \in \boldsymbol{\Omega}} \rho^L(\mathbf{x}, \xi)}$$

Full data term

Smoothness term

- Piecewise rigidity

Total variation (TV) favors
piecewise smooth solutions

Smoothness term

- **Piecewise rigidity**

Total variation (TV) favors piecewise smooth solutions

Decoupling

Translational field

$$\tau : \Omega \to \mathbb{R}^3$$

$$\xi : \Omega \to \mathbb{R}^6$$

$$\omega : \Omega \to \mathbb{R}^3$$

Rotational field

Smoothness term

- **Piecewise rigidity**

Total variation (TV) favors piecewise smooth solutions

Decoupling

$$\xi : \mathbf{\Omega} \rightarrow \mathbb{R}^6$$

Translational field

$$\tau : \mathbf{\Omega} \rightarrow \mathbb{R}^3$$

$$\omega : \mathbf{\Omega} \rightarrow \mathbb{R}^3$$

Rotational field

$$\mathbf{TV}_c(\tau) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \|\nabla \tau(\mathbf{x})\|$$

Channel-by-channel TV

Smoothness term

✪ **Piecewise rigidity**

Total variation (TV) favors piecewise smooth solutions

Decoupling

Translational field

$$\tau : \boldsymbol{\Omega} \to \mathbb{R}^3$$

$$\xi : \boldsymbol{\Omega} \to \mathbb{R}^6$$

$$\omega : \boldsymbol{\Omega} \to \mathbb{R}^3$$

$$\boxed{e^{\hat{\omega}} : \boldsymbol{\Omega} \to SO(3)}$$

Rotational field

$$\mathbf{TV}_c(\tau) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \|\nabla \tau(\mathbf{x})\|$$

Channel-by-channel TV

Smoothness term

- **Piecewise rigidity**

Total variation (TV) favors piecewise smooth solutions

Decoupling

$$\xi : \mathbf{\Omega} \to \mathbb{R}^6$$

Translational field

$$\tau : \mathbf{\Omega} \to \mathbb{R}^3$$

$$\mathbf{TV}_c(\tau) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \|\nabla \tau(\mathbf{x})\|$$

Channel-by-channel TV

$$\omega : \mathbf{\Omega} \to \mathbb{R}^3$$

$$\boxed{e^{\hat{\omega}} : \mathbf{\Omega} \to SO(3)}$$

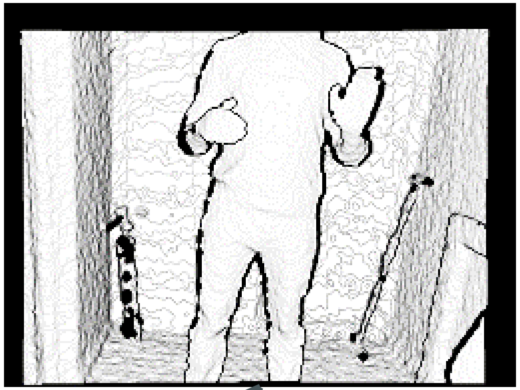
Rotational field

$$\mathbf{TV}_\sigma(\omega) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \sigma_1(\mathbf{D}\omega(\mathbf{x}))$$

Vectorial TV (Goldluecke et al. SIIMS’12)

Smoothness term

- **Piecewise rigidity**



Total variation (TV) favors piecewise smooth solutions

Decoupling

$$\xi : \mathbf{\Omega} \to \mathbb{R}^6$$

Translational field

$$\tau : \mathbf{\Omega} \to \mathbb{R}^3$$

$$\mathbf{TV}_c(\tau) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \|\nabla \tau(\mathbf{x})\|$$

Channel-by-channel TV

$$\omega : \mathbf{\Omega} \to \mathbb{R}^3$$

$$\boxed{e^{\hat{\omega}} : \mathbf{\Omega} \to SO(3)}$$

Rotational field

$$\mathbf{TV}_\sigma(\omega) = \sum_{\mathbf{x}} c(\mathbf{x}) \, \sigma_1(\mathbf{D}\omega(\mathbf{x}))$$

Vectorial TV (Goldluecke et al. SIIMS'12)

Optimization

- The energy is minimized using the **variable splitting** method:

$$\min_{\xi,\chi} \; E_D(\xi) + \frac{1}{2\kappa}\sum_{\mathbf{x}} |\xi(\mathbf{x}) - \chi(\mathbf{x})|^2 + \alpha E_S(\chi)$$

Optimization

- The energy is minimized using the **variable splitting** method:

$$\min_{\xi,\chi} \boxed{E_D(\xi) + \frac{1}{2\kappa}\sum_{\mathbf{x}} |\xi(\mathbf{x}) - \chi(\mathbf{x})|^2} + \alpha E_S(\chi)$$

Regularized weighted least-squares

IRLS with a Gauss-Newton algorithm

Optimization

- The energy is minimized using the **variable splitting** method:

$$\min_{\xi,\chi} E_D(\xi) + \frac{1}{2\kappa}\sum_{\mathbf{x}} |\xi(\mathbf{x}) - \chi(\mathbf{x})|^2 + \alpha E_S(\chi)$$

Regularized weighted least-squares

IRLS with a Gauss-Newton algorithm

Vectorial ROF model

Gradient descent and re-projection on the dual-ROF model

Optimization

- The energy is minimized using the **variable splitting** method:

$$\min_{\xi,\chi} E_D(\xi) + \frac{1}{2\kappa}\sum_{\mathbf{x}} |\xi(\mathbf{x}) - \chi(\mathbf{x})|^2 + \alpha E_S(\chi)$$

Regularized weighted least-squares

Vectorial ROF model

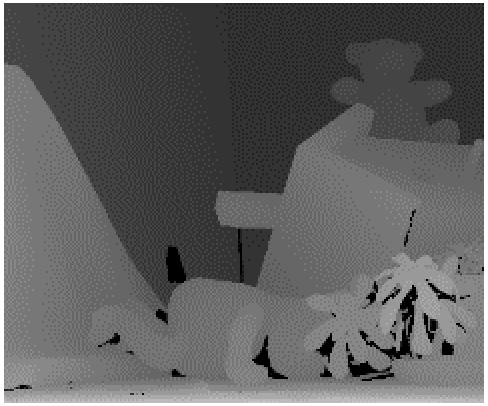
Alternation

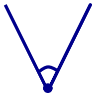
IRLS with a Gauss-Newton algorithm

Gradient descent and re-projection on the dual-ROF model

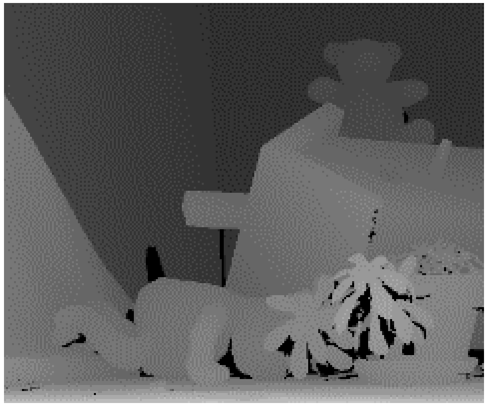
Experiments

Middlebury stereo datasets

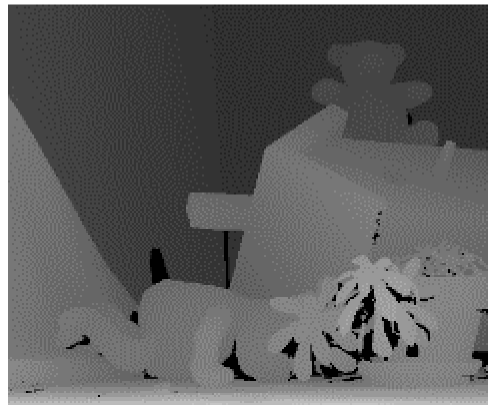


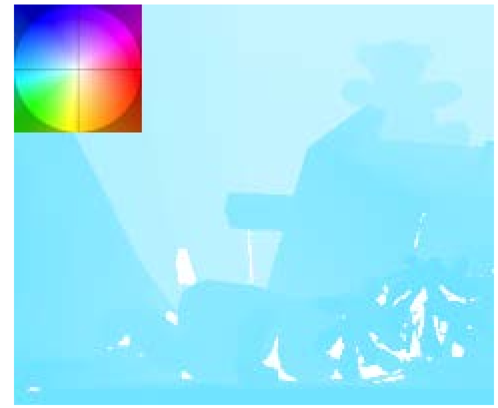


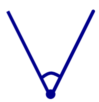
Middlebury stereo datasets



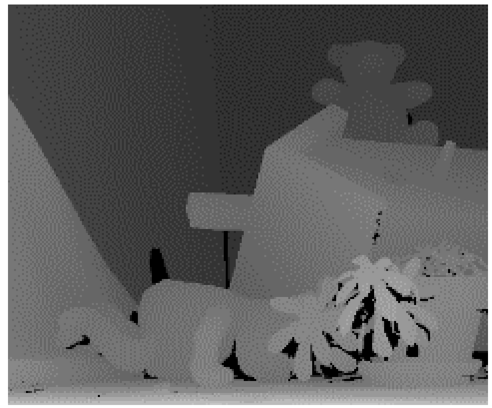
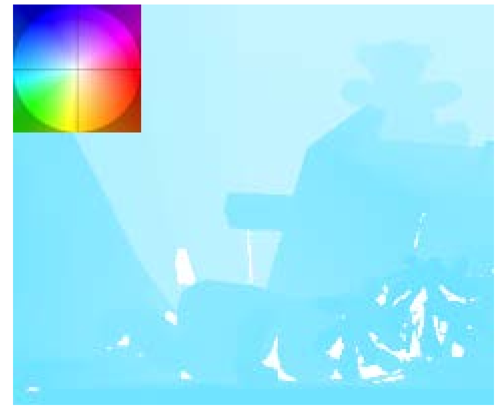
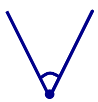
Middlebury stereo datasets





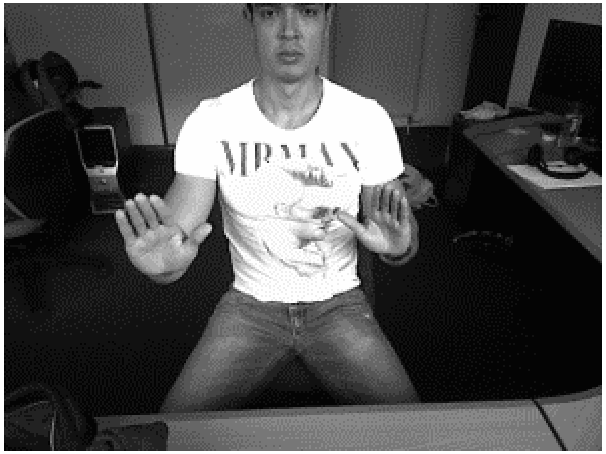


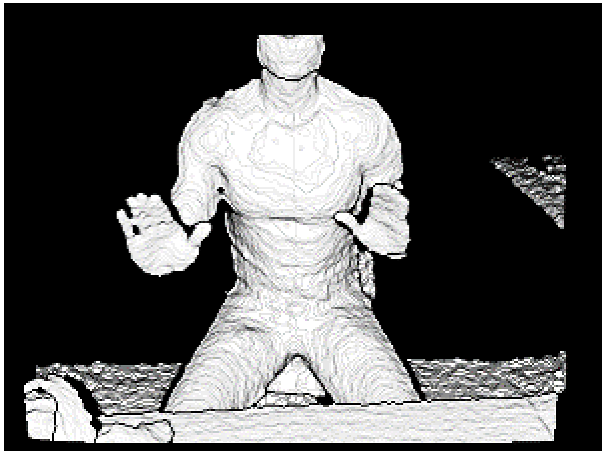
Middlebury stereo datasets

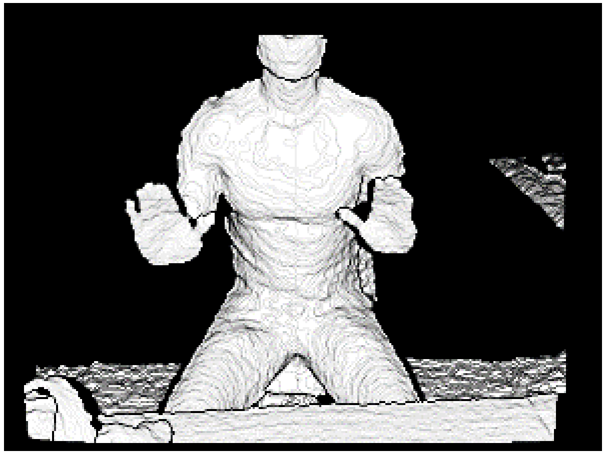
	Views	Teddy		Cones	
		RMS	AAE	RMS	AAE
Semi-rigid Scene Flow (ours)	1	0.49	0.46	0.45	0.37
Hadfield and Bowden [5]	1	0.52	1.36	0.59	1.61
Quiroga *et al.* [13]	1	0.94	0.84	0.79	0.52
Brox and Malik [2] + depth	1	2.11	0.43	2.30	0.52
Basha *et al.* [1]	2	0.57	1.01	0.58	0.39
Huguet and Devernay [8]	2	1.25	0.51	1.10	0.69

Non-rigid motion estimation

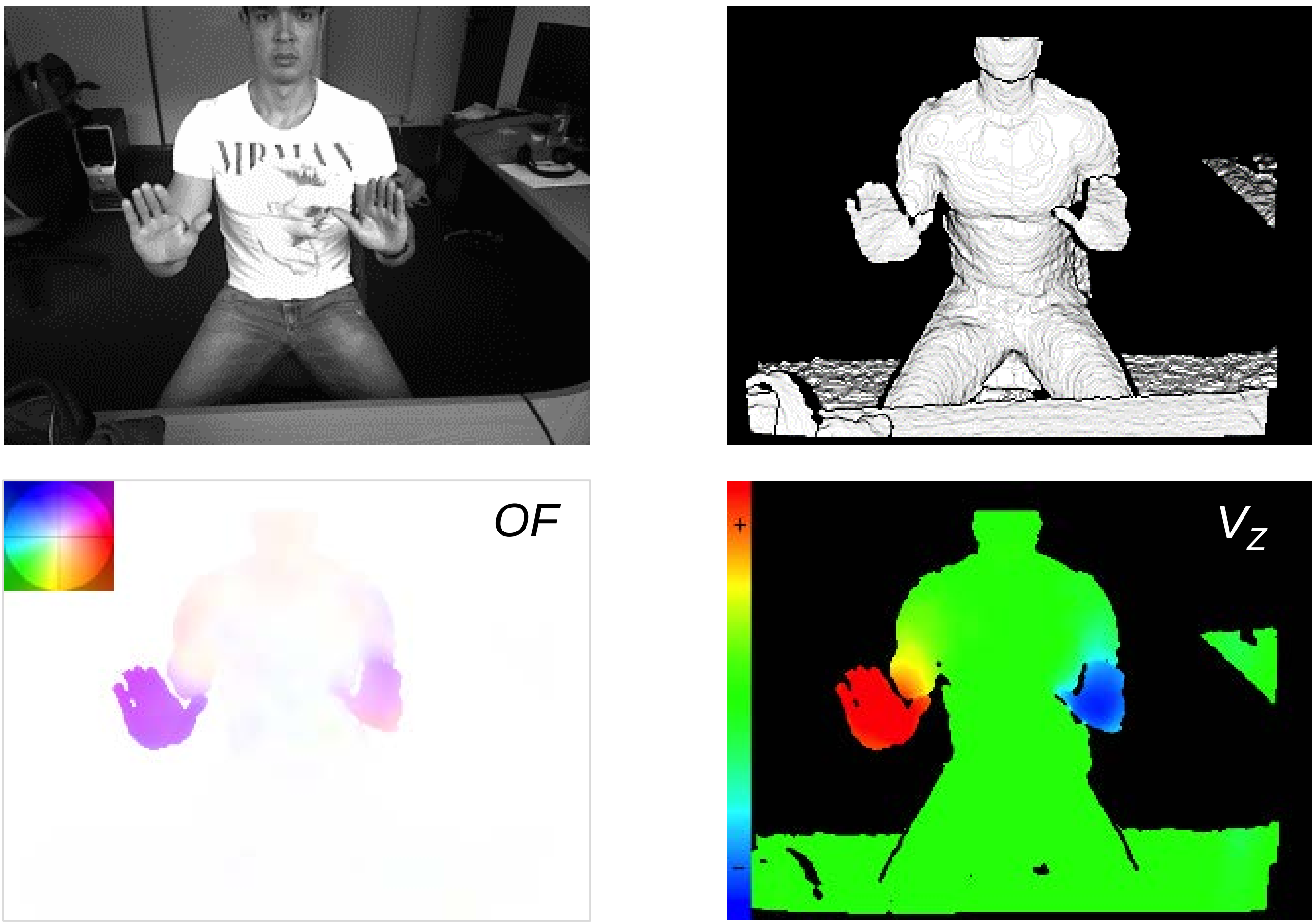




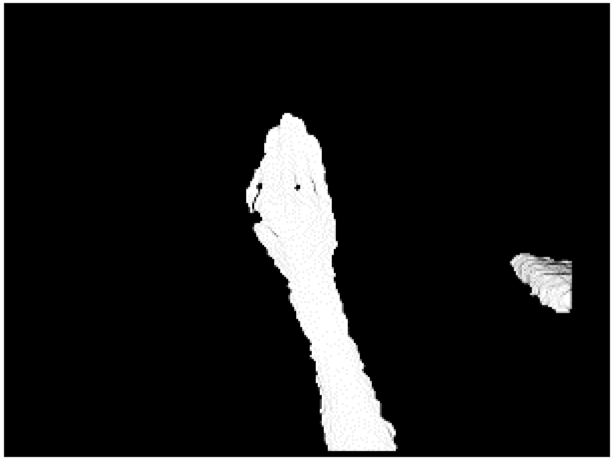
Non-rigid motion estimation



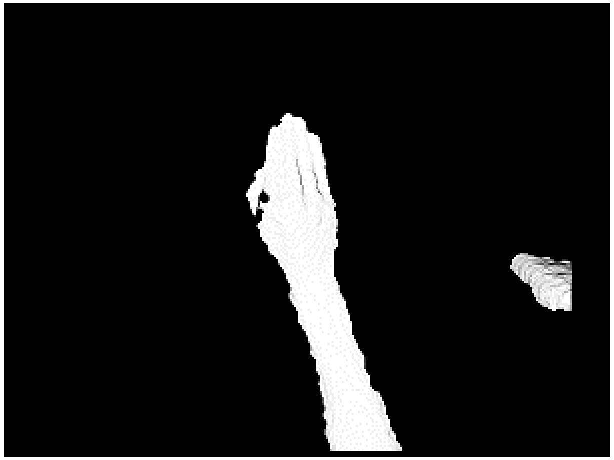
Non-rigid motion estimation



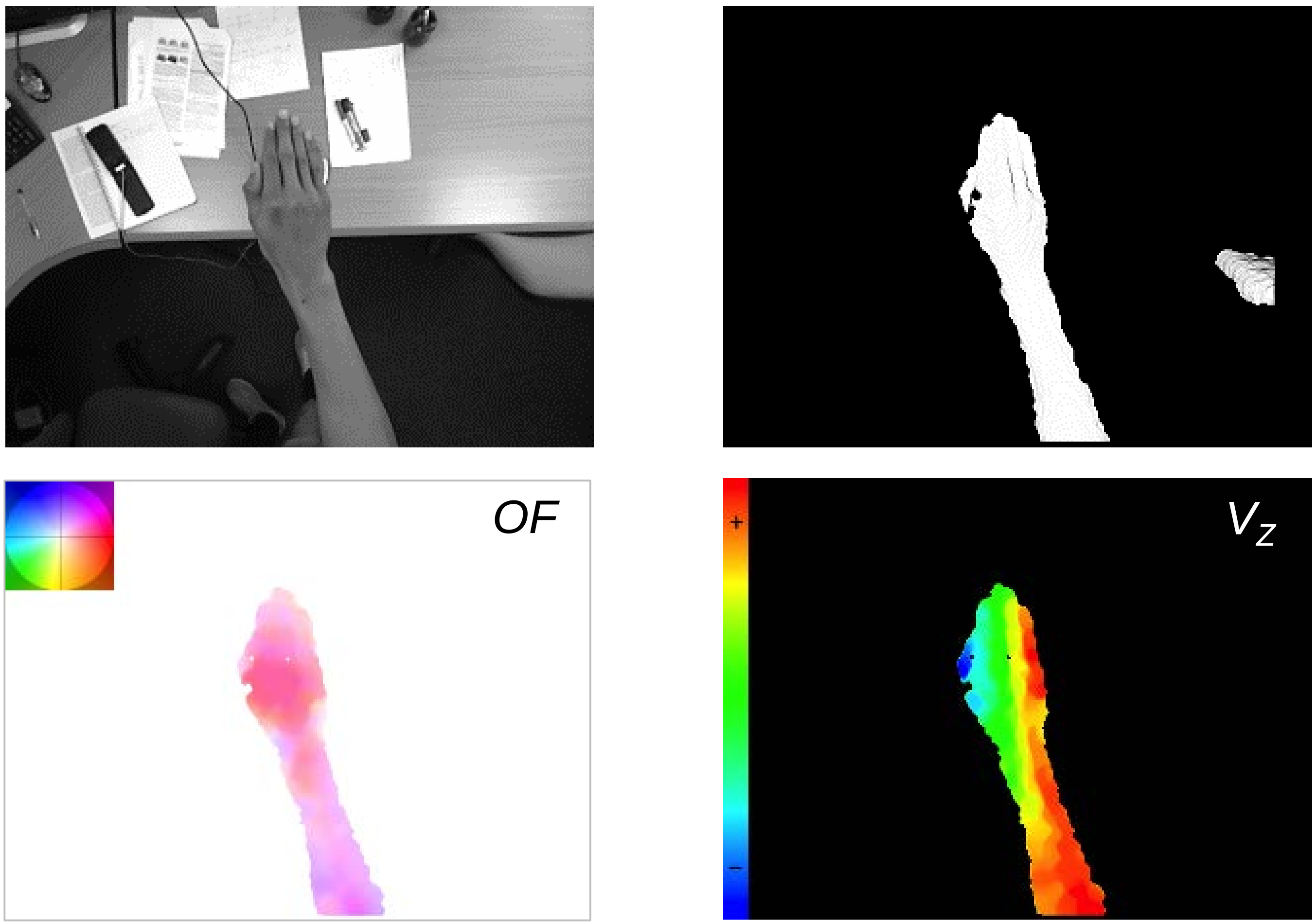
Non-rigid motion estimation (cont.)

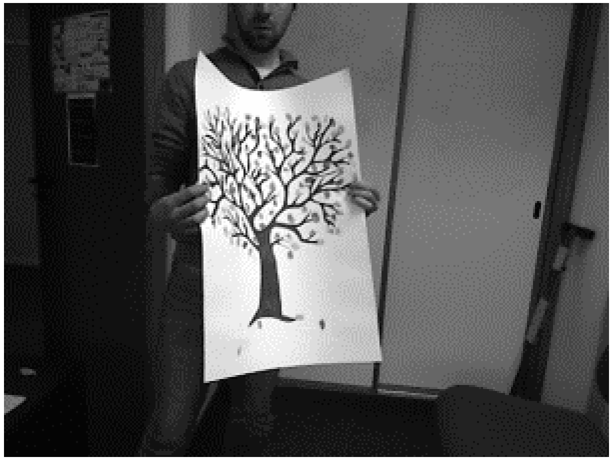
Non-rigid motion estimation (cont.)

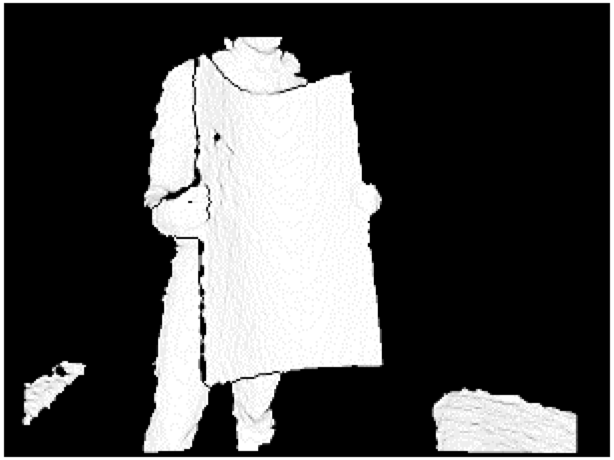


Non-rigid motion estimation (cont.)

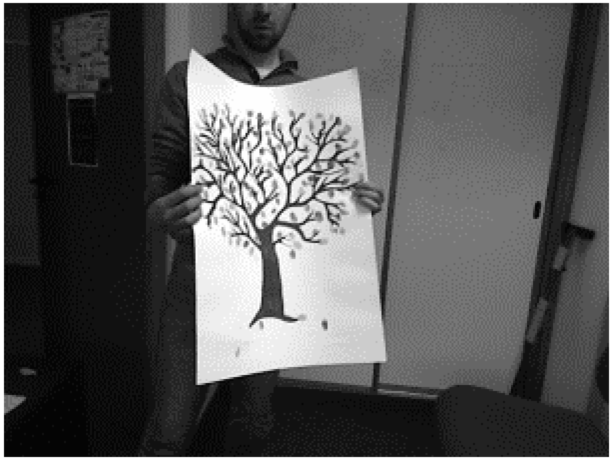


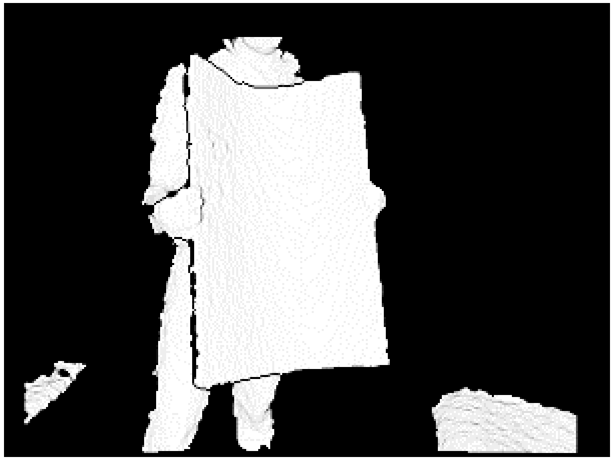
Non-rigid motion estimation (cont.)



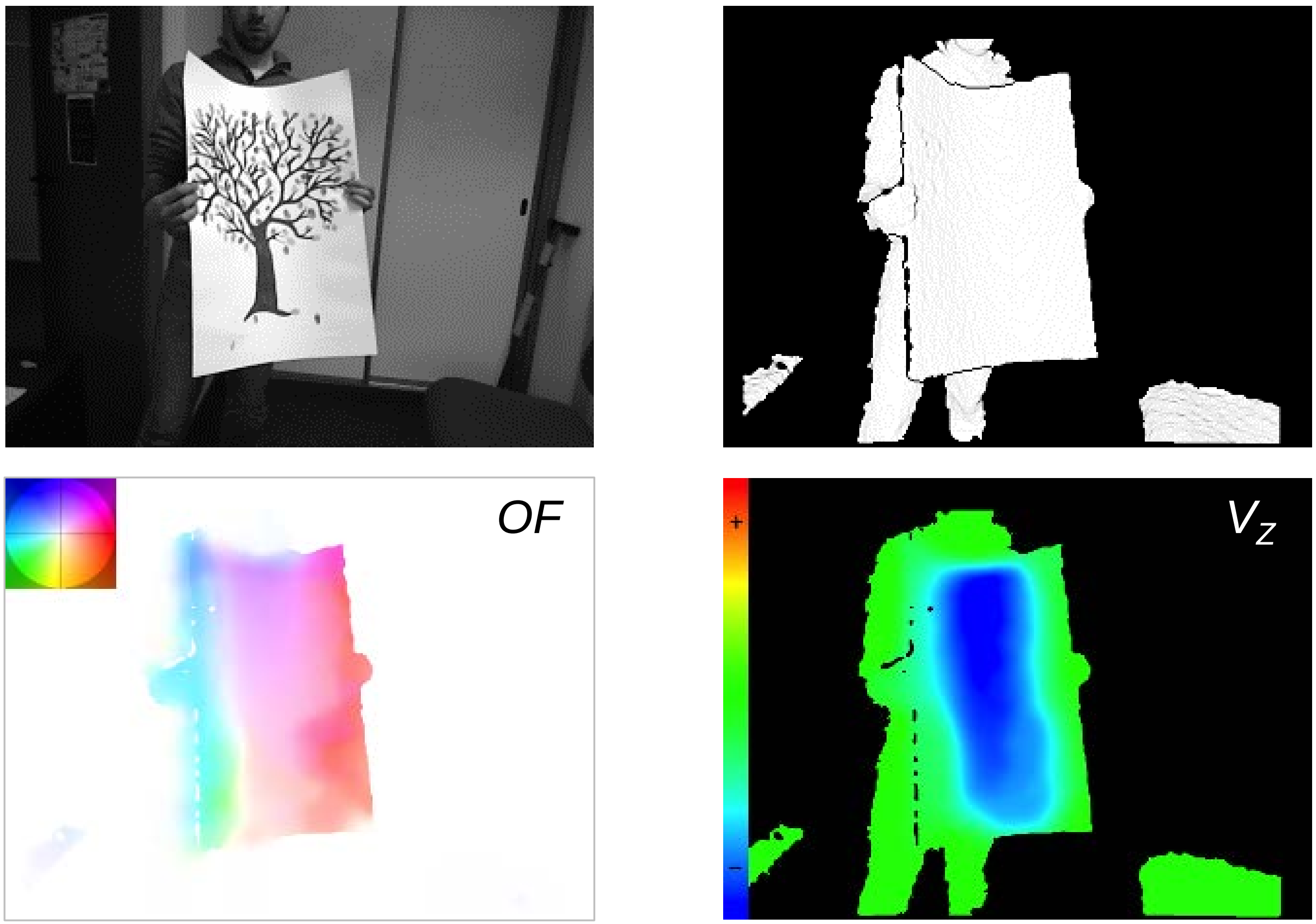


Non-rigid motion estimation (cont.)





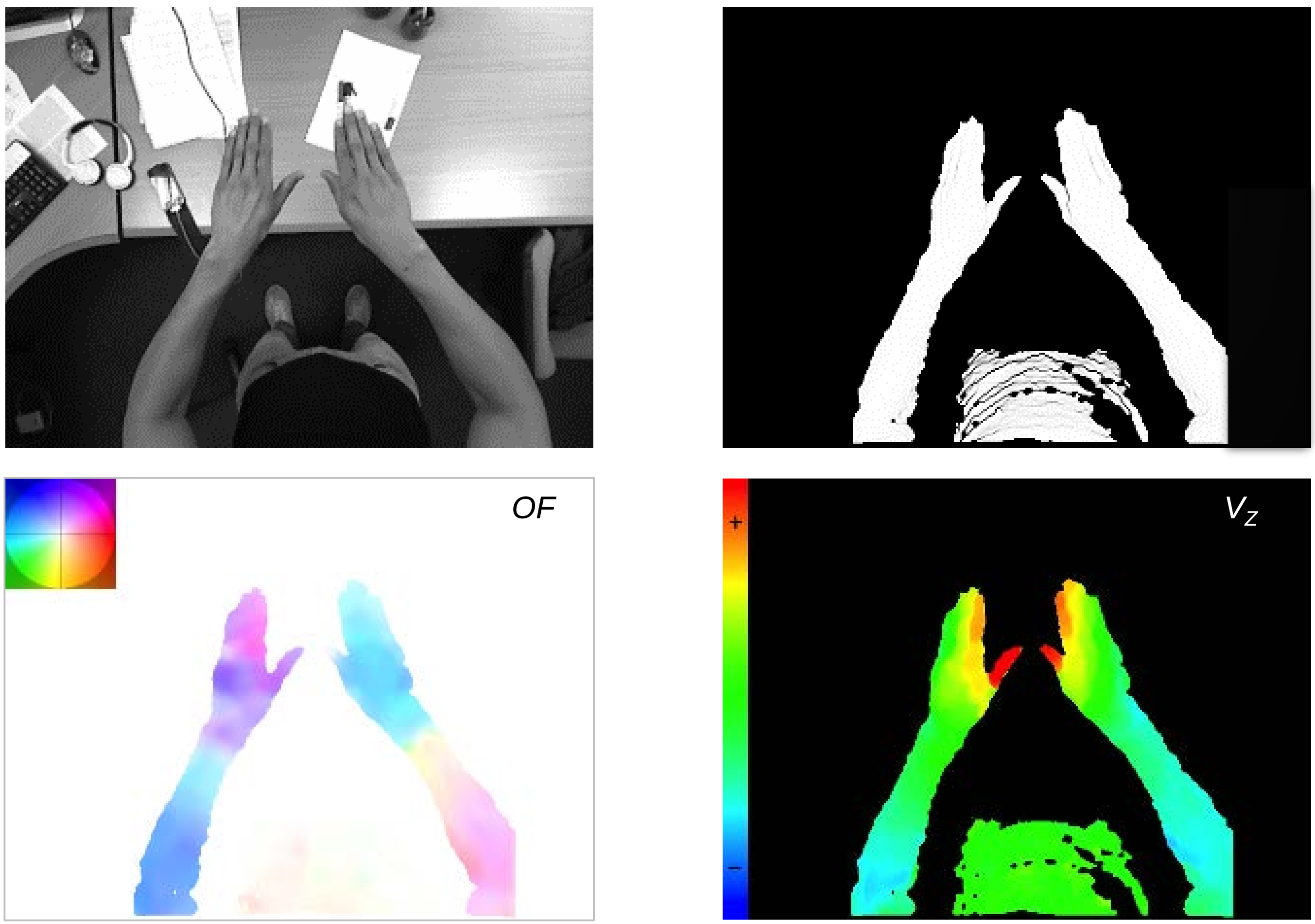
Non-rigid motion estimation (cont.)



Non-rigid motion estimation (cont.)

Non-rigid motion estimation (cont.)

Non-rigid motion estimation (cont.)



Applications

Rigid motion estimation

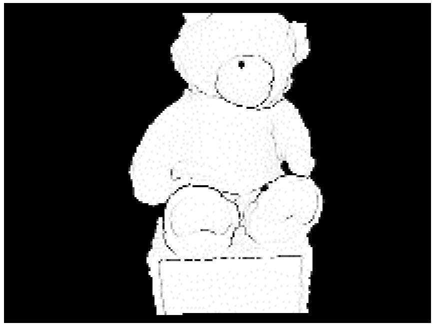
- ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\mathbf{\Omega}} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$

Rigid motion estimation

- ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$

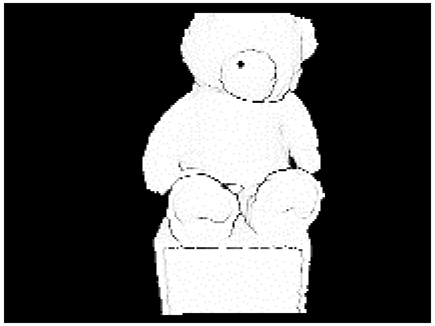


RGBD dataset
(*Sturm et al.* IROS'12)

Rigid motion estimation

✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$



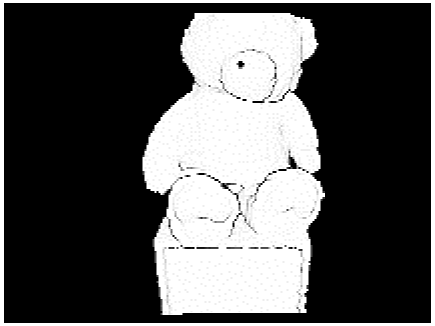
RGBD dataset
(*Sturm et al.* IROS'12)

Rigid motion estimation

✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$

RGBD dataset
(*Sturm et al.* IROS'12)

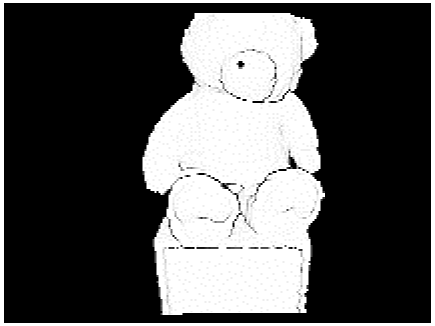


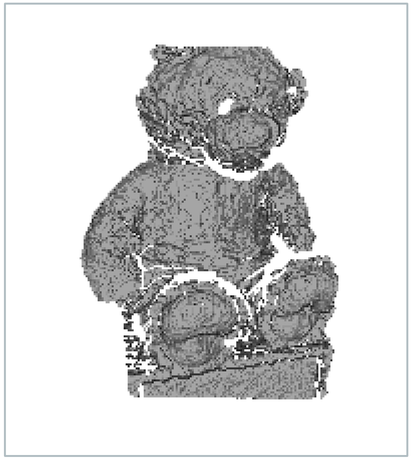
Rigid motion estimation

✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$

RGBD dataset
(*Sturm et al.* IROS'12)

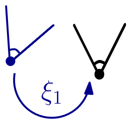




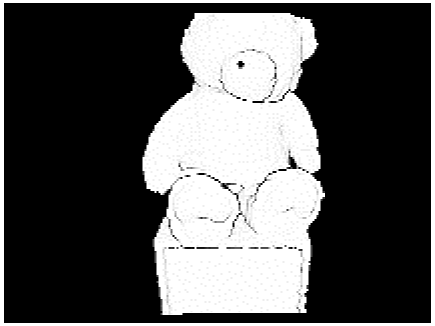
Rigid motion estimation

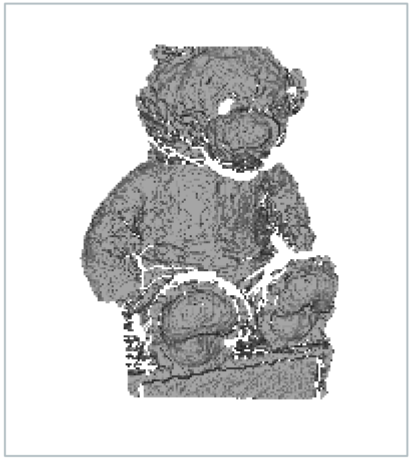
✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$



RGBD dataset
(*Sturm et al.* IROS'12)

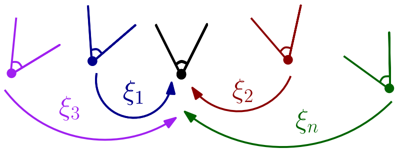




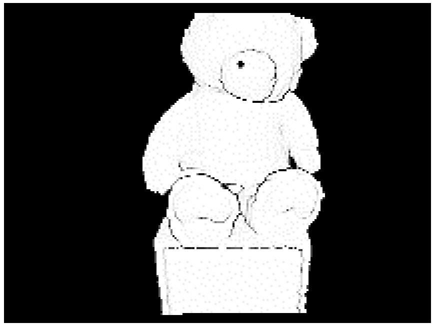
Rigid motion estimation

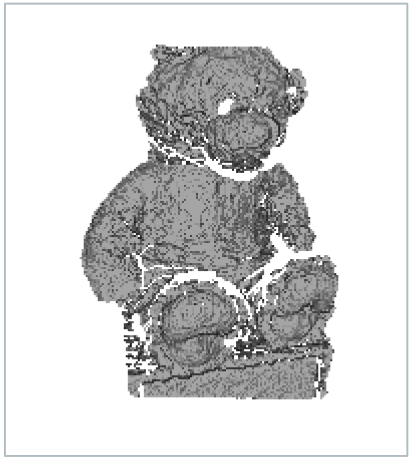
✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$



RGBD dataset
(*Sturm et al.* IROS'12)

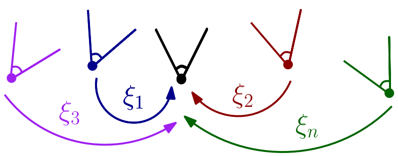


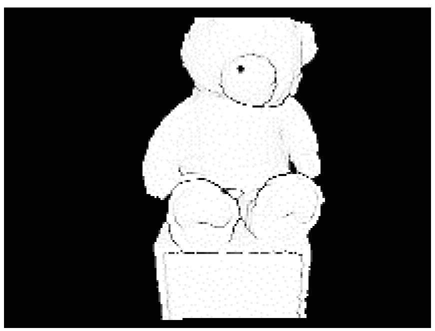


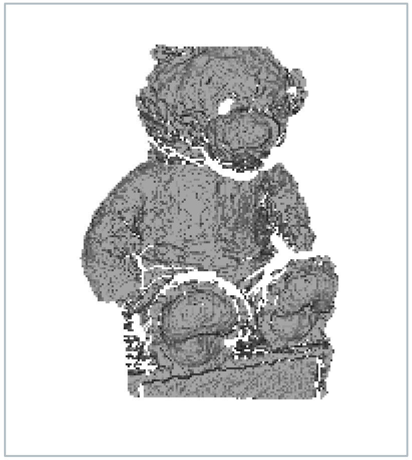
Rigid motion estimation

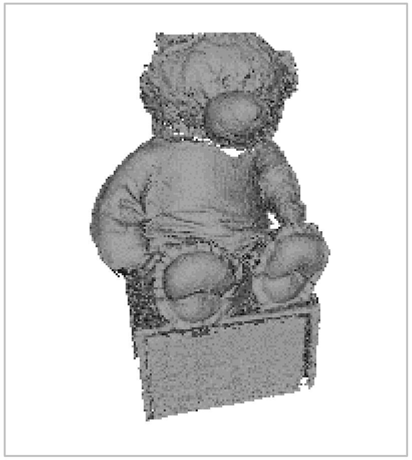
✪ ***A particular case:*** solving for a single twist motion

$$E_{\mathbf{Rig}}(\xi_{\mathrm{R}}) = \sum_{\mathbf{x}\in\Omega} \rho_{data}\left(\mathbf{x}, \xi_{\mathrm{R}}\right)$$









RGBD dataset
(*Sturm et al.* IROS'12)

Scene flow with camera motion estimation

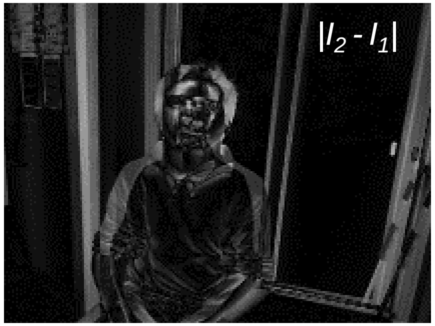
- Knowing the camera motion can simplify the estimation and is useful for applications

Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications

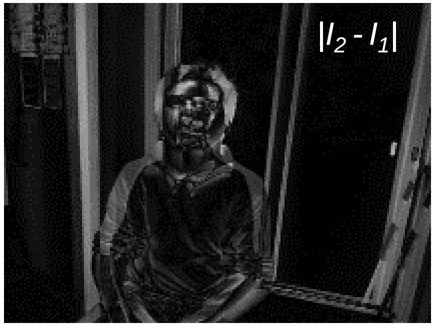
Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications



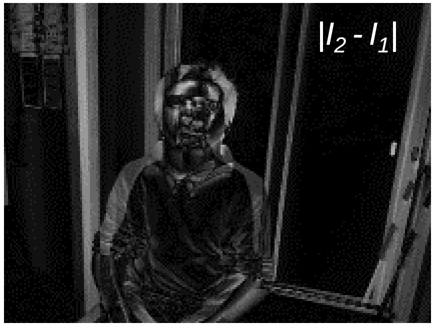
Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications



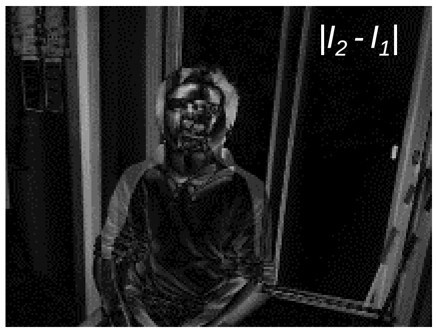
Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications



Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications



Scene flow with camera motion estimation

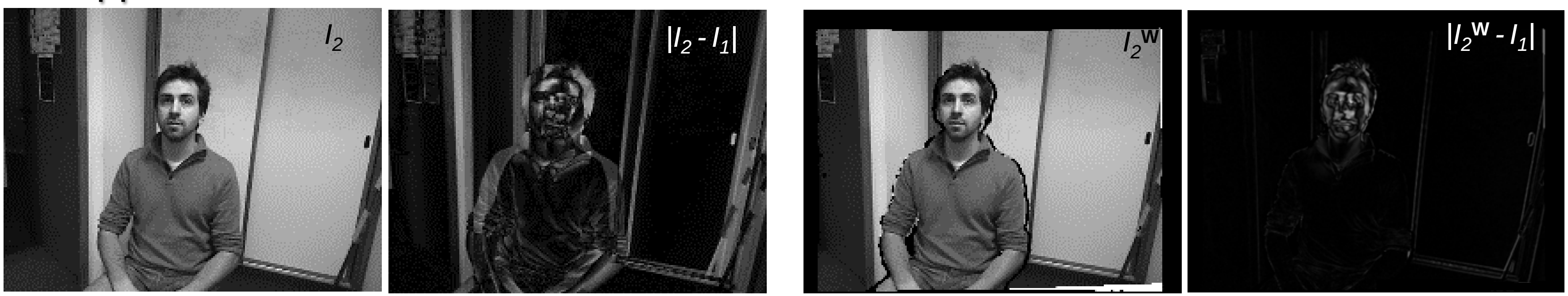
- Knowing the camera motion can simplify the estimation and is useful for applications

- We split the motion field into a **globally rigid component** plus a **non-rigid residual**

$$\chi = \log(e^{\hat{\xi}_{\mathrm{R}}} + e^{\hat{\xi}} - \mathbf{I}_{4\times 4})$$

Scene flow with camera motion estimation

- Knowing the camera motion can simplify the estimation and is useful for applications



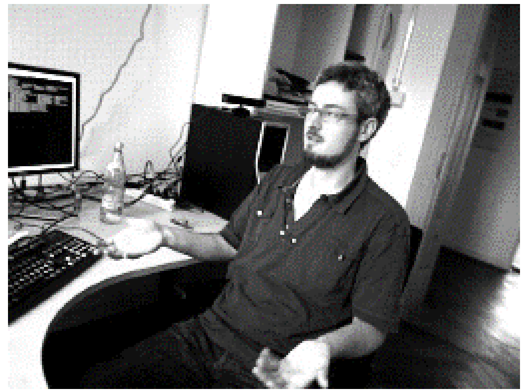
- We split the motion field into a **globally rigid component** plus a **non-rigid residual**

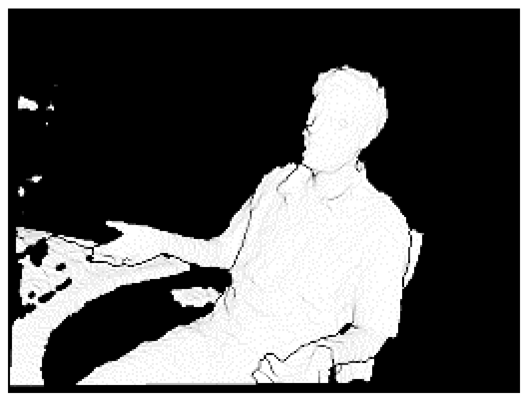
$$\chi = \log(e^{\hat{\xi}_R} + e^{\hat{\xi}} - \mathbf{I}_{4\times 4})$$

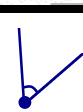
$$\boxed{\min_{\chi} \; E_{\mathbf{Rig}}(\chi) + E_{\mathbf{Res}}(\chi)}$$

Proposed energy

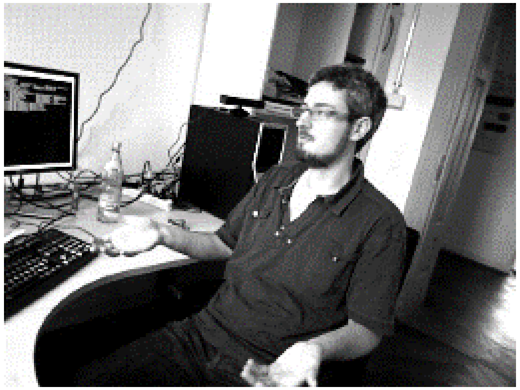
Scene flow with camera motion estimation (cont.)

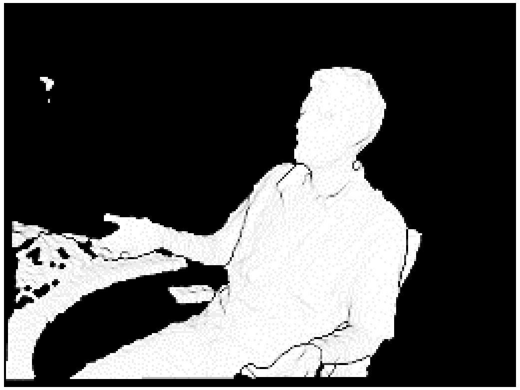


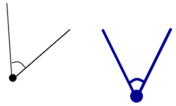




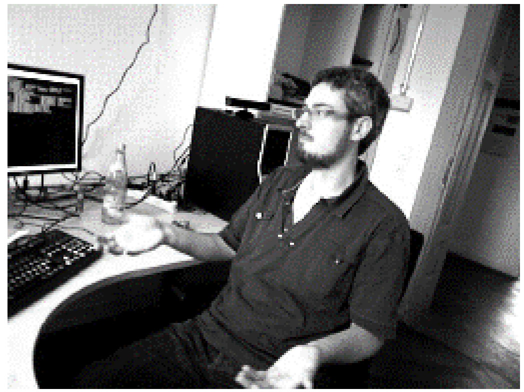
Scene flow with camera motion estimation (cont.)

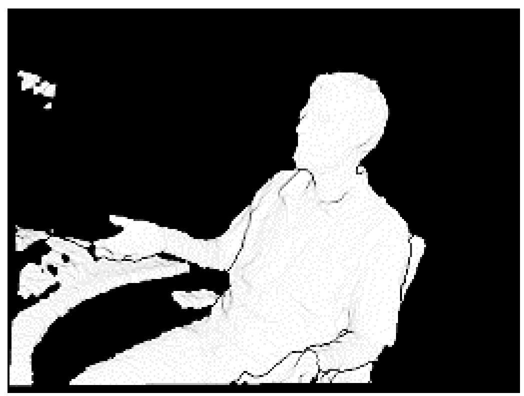


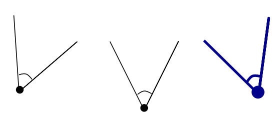




Scene flow with camera motion estimation (cont.)

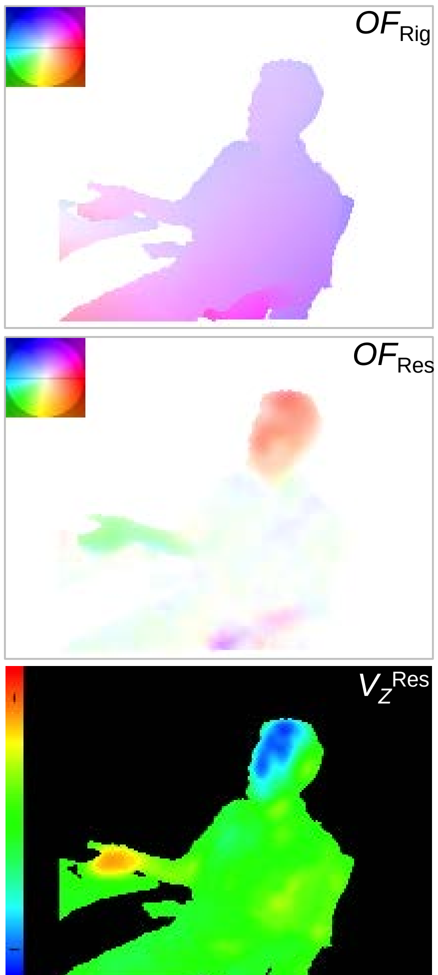


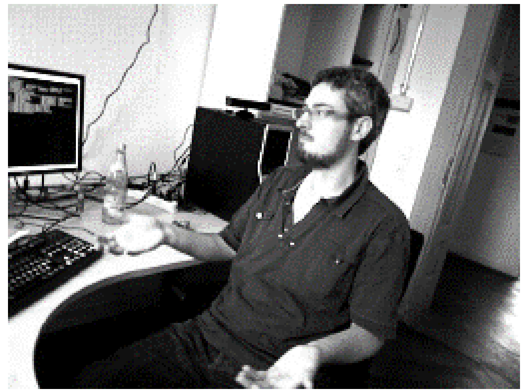


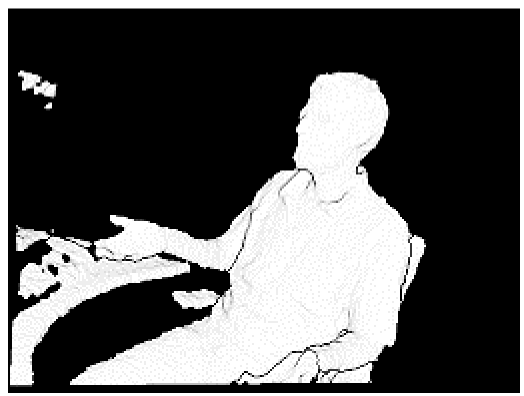


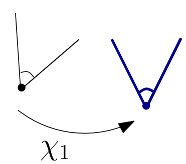
Scene flow with camera motion estimation (cont.)

Scene flow with camera motion estimation (cont.)

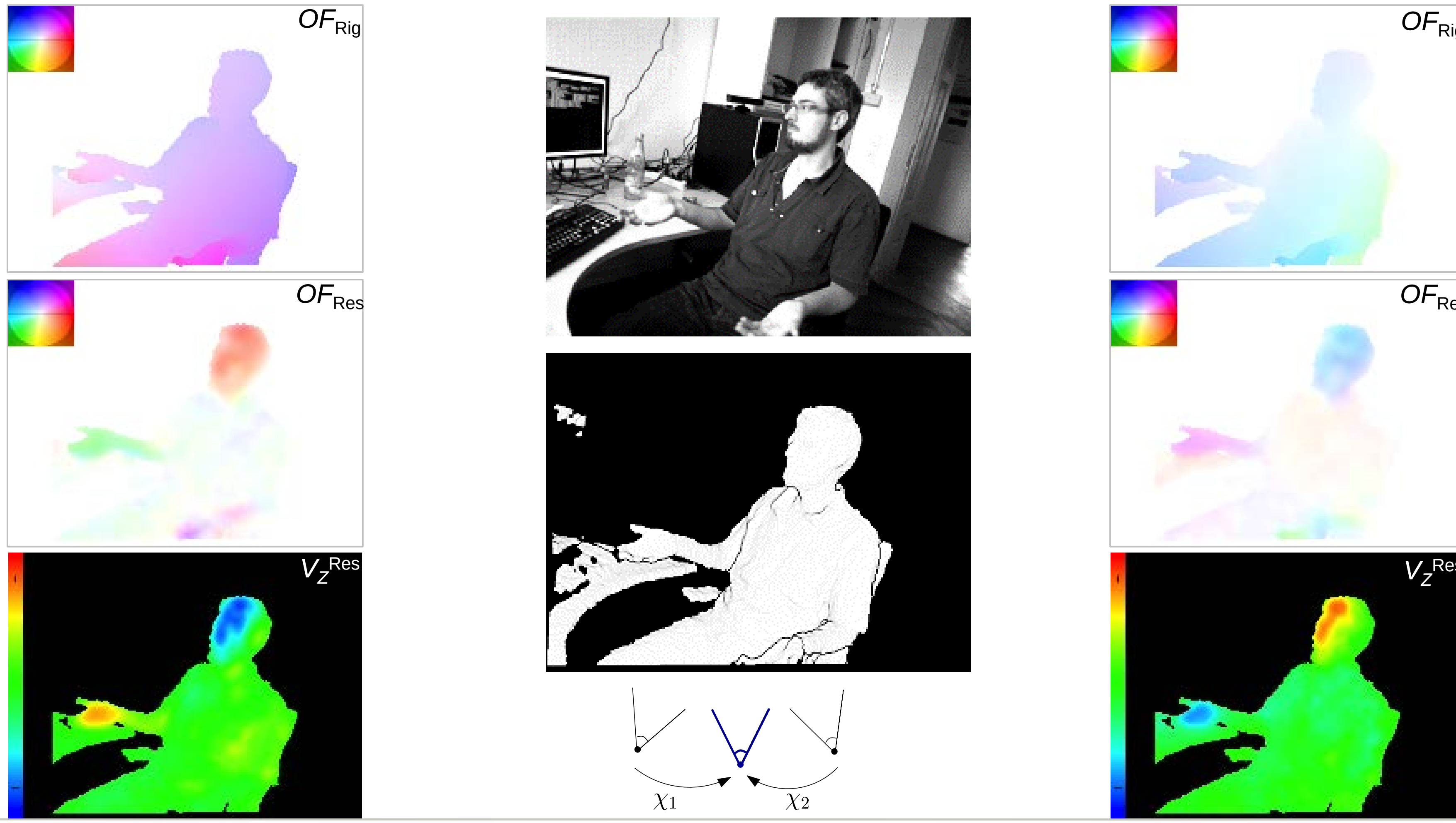








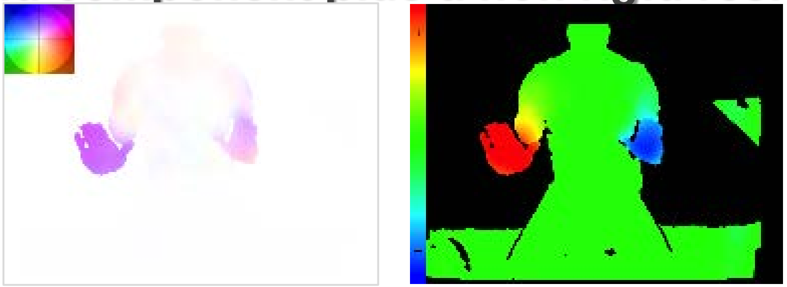
Scene flow with camera motion estimation (cont.)



Conclusion

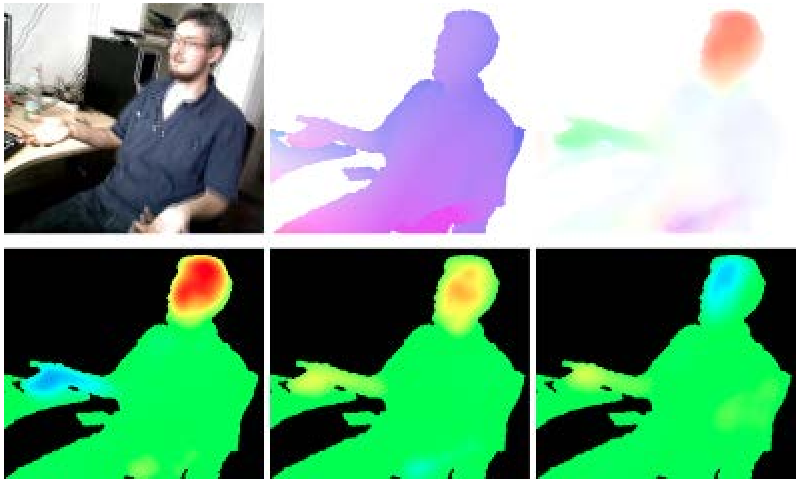
- We have presented a new method to compute a dense scene flow from RGBD images by **modeling the motion as a field of rigid motions**.
- We proposed an **adjustable combination between local and piecewise rigidity**

- We model the motion as a **rigid component plus a non rigid residual**



- **Future work:**
  - Temporal consistency
  - Real-time implementation
  - Large displacements

Thanks!



Semi-rigid scene flow code
will beavailable soon!