Motion Language of Stereo Image Sequence

Tomoya Kato
IMIT, Chiba University
Yayoi-cho 1-33, Inage-ku,
Chiba 263-8522, Japan

Hayato Itoh
IMIT, Chiba University
Yayoi-cho 1-33, Inage-ku,
Chiba 263-8522, Japan
Present Address:
Graduate School of Informatics,
Nagoya University
Furo-cho, Chikusa-ku,
Nagoya 464-8601, Japan
hitoh@mori.m.is.nagoya-u.ac.jp

Atsushi Imiya
IMIT, Chiba University
Yayoi-cho 1-33, Inage-ku,
Chiba 263-8522, Japan
imiya@faculty.chiba-u.jp

Abstract

This paper proposes a method for the symbolisation of temporal changes of the environments around the autonomous robot in a work space using the scene flow field for recognition of events. We first develope a two-phase method for the accurate computation of the scene flow field. Next, we introduce an algorithm to symbolises the motion fields as a string of words using directional statistics. These two operations extract a string of words of motion in front of the robots from stereo images captured by a robot-mounted camera system. We evaluate performance of our method using KITTI scene flow Dataset 2015. The results shows the efficiency of the method for the extraction of events in front of a robot.

1. Introduction

A classical motion analysis method in computer vision is motion tracking, which establishes temporal correspondences of feature points along a sequence of images. In this paper, employing statistical analysis of the temporal scene flow fields computed from a sequence of stereo image pairs, we develop a method to interpret a sequence of images as a sequence of events. Optical flow and scene flow are the planar and spatial motion fields computed from monocular image and stereo image sequences, respectively. These fields describe motion fields in an environment. Our method proposing in this paper evaluates the global smoothness and continuity of motion fields and detects collapses of smoothness of the motion fields in long-time image sequences using transportation of the temporal scene flow field.

In sign language recognition, the extraction of linguistic information from a sequence of images is a fundamental process. Sounds are estimated from a sequence of images in lip-reading. In these two applications, symbolic information such as sounds and the meanings of words and sentences are estimated from a image sequence to recognise and understand visual information as communication purposes.

In the visual navigation of robots, probes and cars, geometric information around them is used for navigation and localisation. Structures in the robot workspace inferred from motion of the robot reconstruct spatial geometry around them. Using the tradition of geometric information around a robot, simultaneously localisation and mapping (SLAM) localises the position of them in an environments.

In ref. [9], using the local stationarity of visual motion, a linear method for motion tracking was introduced. It is possible to extend the local assumption on the optical flow field to higher-order constraints on the motion field. In this paper, we assume that optical flow fields are locally quadratic. For the computation of the three-dimensional scene flow from a stereo image sequence [5, 7, 6], we are required to solve four image-matching problems and their deformation fields. Two of them are optical flow computations for left and right image sequences. The other two of them are stereo matching for two successive stereo pairs. The displacements between stereo pairs are at most locally affine transformations caused by perspective projections based on the camera geometry. The displacement between a pair of successive images in left and right sequences, however, involves higher-order transformations caused by camera motion if a pair of cameras is mounted on a mobile vehicle. Therefore, since we are required to adopt different-order constraints on the optical flow computation and stereo matching for a stereo

pair sequence, we develop a method with locally higher-order constraints for the fast computation of the optical flow field.

The Wasserstein distance defines a metric among probability measures [4]. In computer vision and pattern recognition, the 1-Wasserstein [10] distance is known as the earth movers' distance (EMD). We deal with the distribution of scene flow vectors as directional statistics [1] on the sphere. Then, using the Wasserstein distance for spherical distributions [3], we evaluate the temporal total transportation between a pair of successive scene flow fields.

2. Histogram-based Metric of Vector Field

Setting $\boldsymbol{u}(\boldsymbol{x}) = (u(\boldsymbol{x}), v(\boldsymbol{x}), w(\boldsymbol{x}))^\top$ for $\boldsymbol{x} \in \mathbf{R}^3$ to be a vector field in three-dimensional Euclidean space, the directional histogram of $\boldsymbol{u}$ is integration of the magnitude of $\boldsymbol{u}$ with respect to the direction of $\boldsymbol{u}$, that is,

$$s(\boldsymbol{n}) = \frac{1}{|\Omega|} \int_{\Omega \frac{\boldsymbol{u}}{|\boldsymbol{u}|} = \boldsymbol{n}} |\boldsymbol{u}(\boldsymbol{x})| d\boldsymbol{x}, \tag{1}$$

where Ω and $|\Omega|$ are the region of interest and its area measure, respectively. Since a histogram is a probabilistic distribution, we define the pointwise distance between a pair of histograms defined by eq. (1) as

$$\begin{aligned} & d(\boldsymbol{u}_1, \boldsymbol{u}_2)^2 \\ &= \min_c \int_{S^2} \int_{S^2} |s_1(\boldsymbol{m}) - s_2(\boldsymbol{n})|^2 \\ & \qquad \times c(\boldsymbol{m}, \boldsymbol{n}) d\boldsymbol{m} d\boldsymbol{n}. \end{aligned} \tag{2}$$

If we consider the rotation of histogram, which is achieved by a rotation matrix $\boldsymbol{R}$, the metric becomes

$$\begin{aligned} & d_R(\boldsymbol{u}_1, \boldsymbol{u}_2)^2 \\ &= \min_{c, \boldsymbol{R}} \int_{S^{n-1}} \int_{S^{n-1}} |s_1(\boldsymbol{m}) - s_2(\boldsymbol{R}\boldsymbol{n})|^2 \\ & \qquad \times c(\boldsymbol{m}, \boldsymbol{n}) d\boldsymbol{m} d\boldsymbol{n}. \end{aligned} \tag{3}$$

Equation (2) is the 2-Wasserstein distance for distributions on the unit sphere S^2.

Setting T to be the icosahedral-triangle grid on the unit sphere, the centroid of each triangular face is $\boldsymbol{v}_i$. From directional data $h(\boldsymbol{n})$ for $\boldsymbol{n} \in S^2$, the voting to the triangular bin i whose centroid is $\boldsymbol{v}_i$ is computed as the summation

$$s(i) = \sum_{\boldsymbol{v}_i = arg\min_k \angle(\boldsymbol{v}_k, \boldsymbol{n})} s(\boldsymbol{n}). \tag{4}$$

The Wasserstein distance between two histograms on T is computed by

$$D(s_1, s_2) = \min_{c_{ij}} \sum_{\boldsymbol{v}_i \in V} \sum_{\boldsymbol{v}_j \in V} |s_1(i) - s_2(j)|^2 c_{ij} \tag{5}$$

as a discretisation of the Wasserstein distance for a spherical distribution. Figure 1 shows discretisation of spherical histogram based on eq. (4).

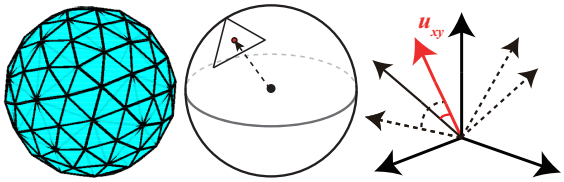
The discrete problem is solved by minimising D,

$$D = \sum_{i,j=1}^{n} a_{ij} x_{ij} \tag{6}$$

$$\text{w.r.t.} \quad \sum_{i=1}^{n} c_i j = s_1(j), \sum_{j=1}^{n} x_{ij} = s_2(i),$$

$$x_{ij} \geq 0, \tag{7}$$

setting $a_{ij} = |s_1(i) - s_2(j)|^2$. We solve this minimisation problem using interior point method.



(a) (b) (c)

Figure 1. Generation of spherical histogram. (a) Geodesic grid generated from an icosahedron. (b) The centroid of a face of the geodesic grid. (c) Voting process based on inner products.

3. Local Optical Flow Computation

3.1. Local Optical Flow Field

Since point correspondences between a pair of stereo images are expressed by an affine transform, we assume that local correspondences between points are affine. On the other hand, we assume that the optical flow field on each image sequence of a stereo image sequence are locally quadratic, since the trajectories of autonomous vehicles are locally smooth and differentiable.

For $f(x, y, t)$, the optical flow vector [8] $\boldsymbol{u} = \dot{\boldsymbol{x}} = (\dot{x}, \dot{y})^\top$, where $\dot{x} = u = u(x, y)$ and $\dot{y} = v = v(x, y)$, of each point $\boldsymbol{x} = (x, y)^\top$ is the solution of the singular equation

$$f_x u + f_y v + f_t = \nabla f^\top \boldsymbol{u} + f_t = 0. \tag{8}$$

In $\Omega(\boldsymbol{c}) = \{\boldsymbol{x} | \ |\boldsymbol{x} - \boldsymbol{c}|_\infty \leq k\}$, for a positive integer k, where $|\boldsymbol{x}|_\infty$ is the l_∞ norm on the plane $\mathbf{R}^2$, setting $\bar{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{c}$, we assume that

$$\boldsymbol{u}_1(\boldsymbol{x}) = \boldsymbol{D}\bar{\boldsymbol{x}} + \boldsymbol{d}. \tag{9}$$

The matrix $\boldsymbol{D}$ and the vector $\boldsymbol{d}$ in eq. (9) minimise the criterion

$$E_1 = \frac{1}{2} \cdot \frac{1}{|\Omega(\boldsymbol{x})|} \int_{\Omega(\boldsymbol{c})} |\nabla f^\top (\boldsymbol{D}\boldsymbol{x} + \boldsymbol{d}) + f_t|^2 d\boldsymbol{x} \tag{10}$$

is the local affine optical flow. The local affine optical flow is the solution of the system of linear equations

$$\boldsymbol{A}_{(1)}\boldsymbol{v}_{(1)} + \boldsymbol{b}_{(1)} = 0 \tag{11}$$

for

$$\begin{aligned} \boldsymbol{A}_{(1)} &= \begin{pmatrix} \boldsymbol{G}, & \boldsymbol{x}^\top \otimes \boldsymbol{G} \\ \boldsymbol{x} \otimes \boldsymbol{G}, & (\boldsymbol{x}\boldsymbol{x}^\top) \otimes \boldsymbol{G} \end{pmatrix}, \\ \boldsymbol{v}_{(1)} &= \begin{pmatrix} \boldsymbol{d} \\ vec\boldsymbol{D} \end{pmatrix}, \ \boldsymbol{b}_{(1)} = \begin{pmatrix} \boldsymbol{a} \\ \boldsymbol{x} \otimes \boldsymbol{a} \end{pmatrix}, \end{aligned} \tag{12}$$

where

$$\boldsymbol{G} = \frac{1}{|\Omega(\boldsymbol{c})|}\int_{\Omega(\boldsymbol{x})} \nabla f \nabla f^\top d\boldsymbol{x}, \tag{13}$$

and

$$\boldsymbol{a} = \frac{1}{|\Omega(\boldsymbol{c})|}\int_{\Omega(\boldsymbol{x})} f_t \nabla f d\boldsymbol{x}. \tag{14}$$

Next, we assume that

$$\boldsymbol{u}_2(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} \\ \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} \end{pmatrix} + \boldsymbol{D}\boldsymbol{x} + \boldsymbol{d} \tag{15}$$

for the piecewise-quadratic vector field, with 2×2 symmetric matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, a 2×2 matrix $\boldsymbol{D}$ and a two-dimensional vector $\boldsymbol{d}$, where $\boldsymbol{e} = (1,1)^\top$.

The local quadratic optical flow is the minimiser of

$$E_2 = \frac{1}{2|\Omega(\boldsymbol{x})|}\int_{\Omega(\boldsymbol{x})} \left|\nabla f^\top \boldsymbol{u}_2(\boldsymbol{x}) + f_t\right|^2 d\boldsymbol{y} \tag{16}$$

for $\boldsymbol{u}_2$ defined by eq. (15).

The local quadratic optical flow is the solution of the the system of linear equation

$$\boldsymbol{A}_{(2)}\boldsymbol{v}_{(2)} + \boldsymbol{b}_{(2)} = 0 \tag{17}$$

at each point $\boldsymbol{x}$ where

$$\boldsymbol{A}_{(2)} = \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \\ \boldsymbol{a}_3 \end{pmatrix} \tag{18}$$

for

$$\begin{aligned} \boldsymbol{a}_1 &= (\boldsymbol{G}, \boldsymbol{x}^\top \otimes \boldsymbol{G}, \boldsymbol{x}_\otimes^\top \otimes \boldsymbol{G}\boldsymbol{X}_\otimes^\top), \\ \boldsymbol{a}_2 &= (\boldsymbol{x} \otimes \boldsymbol{G}, (\boldsymbol{x}\boldsymbol{x}^\top) \otimes \boldsymbol{G}, (\boldsymbol{x}\boldsymbol{x}_\otimes^\top) \otimes \boldsymbol{G}\boldsymbol{X}_\otimes^\top), \\ \boldsymbol{a}_3 &= (\boldsymbol{x}_\otimes \otimes \boldsymbol{X}_\otimes\boldsymbol{G}, (\boldsymbol{x}_\otimes\boldsymbol{x}^\top) \otimes \boldsymbol{X}_\otimes\boldsymbol{G}, \\ &\qquad (\boldsymbol{x}_\otimes\boldsymbol{x}_\otimes^\top) \otimes \boldsymbol{X}_\otimes\boldsymbol{G}\boldsymbol{X}_\otimes^\top) \end{aligned}$$

and

$$\boldsymbol{v}_{(2)} = \begin{pmatrix} \boldsymbol{d} \\ vec\boldsymbol{D} \\ vec\boldsymbol{C} \end{pmatrix}, \tag{19}$$

$$\boldsymbol{b}_{(2)} = \begin{pmatrix} \boldsymbol{a} \\ \boldsymbol{x} \otimes \boldsymbol{a} \\ ((\boldsymbol{x}\boldsymbol{x}_\otimes^\top) \otimes \boldsymbol{X})\boldsymbol{a} \end{pmatrix}, \tag{20}$$

where

$$\begin{aligned} \boldsymbol{C} &= Diag(\boldsymbol{P}, \boldsymbol{Q}), && (21) \\ \boldsymbol{x}_\otimes &= \boldsymbol{e} \otimes \boldsymbol{x}, && (22) \\ \boldsymbol{X}_\otimes &= \boldsymbol{I} \otimes \boldsymbol{x} && (23) \end{aligned}$$

for the 2×2 identity matrix $\boldsymbol{I}$. We call the optical flow computation by minimizing E_2 the quadratic KLT (QKLT).

Setting f_L and f_R to be left and right image sequences, respectively, for optical flow vector $\boldsymbol{u} = (u,v)^\top = (\dot{x}, \dot{y})^\top$ on the left image and disparities of the pair of stereo images d and d' for time t and $t+1$, respectively, the relations

$$\begin{aligned} &f_L(x+d, y, t) = f_R(x, y, t), \\ &f_R(x+u+d', y+v, t+1) = f_R(x, y, t), \\ &f_L(x+u, y+v, t+1) = f_L(x, y, t), \end{aligned} \tag{24}$$

are satisfied. From the solutions of eq. (24), the scene flow vector $(X', Y', Z')^\top$ at the time t is computed as

$$\begin{aligned} \dot{\boldsymbol{X}} &= \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} \\ &= b \begin{pmatrix} \frac{(x+u)}{d+d'} - \frac{x}{d} \\ \frac{(y+v)}{d+d'} - \frac{y}{d} \\ \frac{f}{d} - \frac{f}{d+d'} \end{pmatrix}, \end{aligned} \tag{25}$$

where b is the base-length between stereo pairs.

3.2. l_2^2-l_2 Optimisation

For the computation of the local optical flow field, we deal with the minimisation of the functional

$$J_{2221}(\boldsymbol{x}) = \frac{1}{2}|\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}|_2^2 + \lambda|\boldsymbol{x}|_2, \tag{26}$$

where we set $\boldsymbol{A} := \boldsymbol{G}_{(1)} \backslash \boldsymbol{G}_{(2)}$, $\boldsymbol{x} := \boldsymbol{v}_{(1)} \backslash \boldsymbol{v}_{(2)}$ and $\boldsymbol{b} := \boldsymbol{b}_{(1)} \backslash \boldsymbol{b}_{(2)}$. Since the functional derivative of J_{2221} with respect to $\boldsymbol{x}$ is

$$\frac{\delta J_{2221}(\boldsymbol{x})}{\delta \boldsymbol{x}} = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}) + \lambda\frac{\boldsymbol{x}}{|\boldsymbol{x}|_2}, \tag{27}$$

the minimiser of eq. (26) is the solution of

$$(\boldsymbol{A}^\top\boldsymbol{A} + \frac{\lambda}{|\boldsymbol{x}|_2}\boldsymbol{I})\boldsymbol{x} = \boldsymbol{A}^\top\boldsymbol{b}. \tag{28}$$

We compute the solution of eq. (28) using the iteration form

$$\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x}^{(n)} = \boldsymbol{b}^{(n)}, \ \ \boldsymbol{b}^{(n)} = \boldsymbol{A}^\top\boldsymbol{b} - \frac{\lambda}{|\boldsymbol{x}^{(n-1)}|_2}\boldsymbol{x}^{(n-1)} \tag{29}$$

until $|\boldsymbol{x}^{(n+1)} - \boldsymbol{x}^{(n)}|_2 < \epsilon$, where

$$\boldsymbol{x}^{(n)} = (\boldsymbol{A}^\top\boldsymbol{A} - \lambda^{(n)}\boldsymbol{I})^{-1}\boldsymbol{A}^\top\boldsymbol{b}, \ \lambda^{(n)} = \frac{\lambda}{|\boldsymbol{x}^{(n-1)}|}. \tag{30}$$

This procedure is performed in Algorithm 1. In this algorithm, $\boldsymbol{x}(i)$ expresses the ith element of vector $\boldsymbol{x}$.

Algorithm 1: $l_2^2 - l_2$ Minimisation

Data: $\boldsymbol{x}^0 := \mathbf{1}, k := 0, 0 \leq \delta \ll 1, 0 < \epsilon$
Result: minimiser of $\frac{1}{2}|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}|_2^2 + \lambda|\boldsymbol{x}|_2$
while $|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}|_2 > \delta$ **do**
  $\lambda^{(k-1)} := \frac{\lambda}{|\boldsymbol{x}^{(k-1)}|_2}$;
  solve $(\boldsymbol{A}^\top \boldsymbol{A} + \lambda^{(k)} \boldsymbol{I})\boldsymbol{x}^{(k)} = \boldsymbol{A}^\top \boldsymbol{b}$;
  if $\boldsymbol{x}^{(k)}(i) = 0$ **then**
    $\boldsymbol{x}^{(k)}(i) := \boldsymbol{x}^{(k)}(i) + \epsilon$;
  end
  $k := k + 1$
end

We call this method of the optical flow computation the $l_2^2 - l_2$ Kanade-Lucas-Tomasi tracker (2221KLT tracker). Moreover, to ensure stable and robust computation, we use the pyramid-transform-based [11, 12] multiresolution method. We call the methods based on algorithms 2 and 3 the pyramid-based 2221Affine KLT (2221PAKLT) and pyramid-based 2221Quadratic KLT (2221PQKLT) trackers, respectively.

Algorithm 2: Affine-Optical-Flow Computation with Gaussian Pyramid

Data: $\boldsymbol{u}^{L+1} := 0, L \geq 0, l := L$
Data: $f_k^L \cdots f_k^0$
Data: $f_{k+1}^L \cdots f_{k+1}^0$
Result: optical flow u_k^0
while $l \geq 0$ **do**
  $f_{k+1}^l := f_{k+1}^l(\cdot + E(\boldsymbol{u}_k^{l+1}), k+1)$;
  compute $\boldsymbol{D}_k^l$ and $\boldsymbol{d}_k^l$;
  $\boldsymbol{u}_k^l := \boldsymbol{D}_k^l \boldsymbol{x}^l + \boldsymbol{d}_k^l$;
  $l := l - 1$
end

Algorithm 3: Quadric-Optical-Flow Computation with Gaussian Pyramid

Data: $\boldsymbol{u}^{L+1} := 0, L \geq 0, l := L$
Data: $f_k^L \cdots f_k^0$
Data: $f_{k+1}^L \cdots f_{k+1}^0$
Result: optical flow u_k^0
while $l \geq 0$ **do**
  $f_{k+1}^l := f_{k+1}^l(\cdot + E(\boldsymbol{u}_k^{l+1}), k+1)$;
  compute $\boldsymbol{C}_k^l$, $\boldsymbol{D}_k^l$ and $\boldsymbol{d}_k^l$;
  $\boldsymbol{u}_k^l := \boldsymbol{x}_1^{l\top} \boldsymbol{C}_k^l \boldsymbol{x}_1^l + \boldsymbol{D}_k^l \boldsymbol{x}^l + \boldsymbol{d}_k^l$;
  $l := l - 1$
end

4. Transportation and Symbolisation of Motion

We define the coherency of motion along the time axis and in a scene. Then, we introduce a measure for the evaluation of the coherency of motion in an image sequence.

Definition 1 *If a vector field on an image generated by the motion of a scene and moving objects is spatially and temporally smooth, we call this property of the field motion coherency.*

Therefore, rapid changes in the spatial direction of motion cause a collapse of motion coherency on the imaging plane, even if the spatial motion of the object is smooth. the sudden halting of a moving object destroys motion smoothness and causes the collapse of the motion coherency.

Setting h_t and h_{tt} to be the first and second derivatives, respectively, of the histogram h, we define the interval $I_i = [t_i, t_{i+1}]$ along the time axis t using a pair of successive points for extremals $h_{tt} = 0$. Using the l_1 linear approximation of h such that

$$\bar{h}(t) = a_i t + b_i, \tag{31}$$

which minimises the criterion

$$J(a_i, b_i) = \sum_{i=1}^{n} \sum_{j=1}^{n(i)} |h(t_{i(j)}) - (a_i t_{i(j)} + b_i)|, \tag{32}$$

where $t_{i(j)} \in I_i$, we allocate signs for spatial motion.

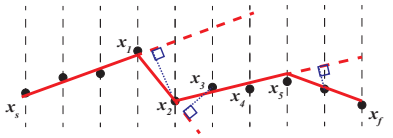


Figure 2. Example of l_1-approximation. $\{x_i\}_{i=1}^5$ are extremals of the interpolated piecewise-linear curve. The procedure detects extremals as knots of a piecewise linear-approximation.

We derive an on-line method for the piecewise interpolation of sample points. Setting $\{x_i\}_{i=1}^k$ to be the sample points of $y = h(x)$, the l_1 approximation of the string of points using the following procedure.

1. For the smallest endpoint x_s, detect extremal (x, y) which minimises $|x_s - y|$.
2. On $[x_s,\ y]$, compute l_1 approximation line, and set it l.
3. If $d(x_i, l) \geq \epsilon$, then $x_s := x_i$.
4. go to step 2.

On the interval $[x_i, x_{i+1}]$, we compute a_i and b_i which minimise

$$T = \sum_{i=1}^{n} w_i \tag{33}$$

$$\text{w.r.t.} \quad -w_i \leq y_i - (ax_i + b) \leq w_i, \quad w_i \geq 0. \tag{34}$$

This procedure detects extremals as knots of a piecewise-linear approximation.

From the sign of a_i, we define the symbols of motion in the interval I_i as $\{\nearrow, \rightarrow, \searrow\}$, where

$$S(h) = \begin{cases} \nearrow & \text{if } a_i > 0 \text{ if } t \in I_i, \\ \rightarrow & \text{if } a_i = 0 \text{ if } t \in I_i, \\ \searrow & \text{if } a_i < 0 \text{ if } t \in I_i. \end{cases} \tag{35}$$

Table 1 lists the environments around an autonomous driving car on the streets in a city and a suburban area. Equation (35) transforms the descriptions in Table 1 to a sequence of words using linear approximation of $h(t)$.

If the motion is smooth in all directions, the time trajectory of the inter-frame Wasserstein distance is constant. If a robot or car is turning, the vector field of scene flow changes the mean direction of the field. Therefore, the transportation of the vector field of scene flow first decreases and second increases. This property of temporal transportation of the vector field appears as a V-shaped variations in the temporal trajectory of the inter-frame Wasserstein distance. Therefore, using a piecewise-linear approximation of the inter-frame Wasserstein distance, it is possible to define the motion geometry from a sequence of images captured by a vehicle-mounted camera.

Since the motion field vectors are small for the estimation of scene flow field in background scene, points in the background are segmented from moving parts using the lengths of the optical flow vectors on each plane of the stereo image sequence. In this step, the motion fields of both left and right image sequences using the KLT. The optical flow vector $\boldsymbol{u}_K$ of each point computed by the KLT, we set

$$\boldsymbol{u}_1 = \begin{cases} \boldsymbol{u}_K, & \text{if } |\boldsymbol{u}_K| \leq 0.4, \\ 0, & \text{otherwise}, \end{cases} \tag{36}$$

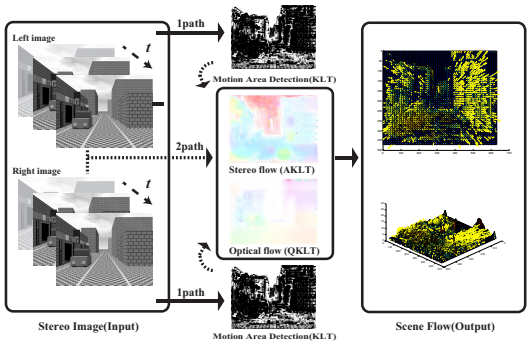
as the results of the first step. Then, in the second step, we compute scene flow for the region R_1, such that

$$R_1 = \{(x, y)^\top | \boldsymbol{u}_1 \neq 0\}. \tag{37}$$

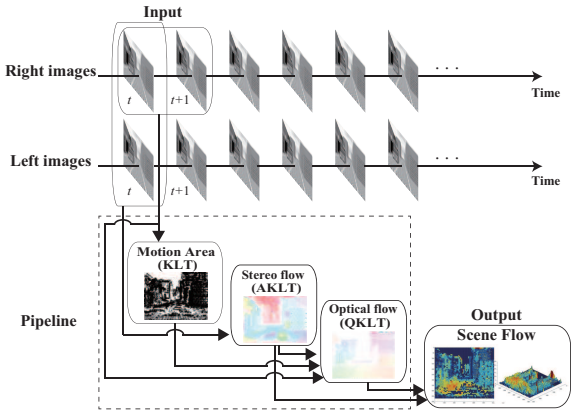
Figure 3(a) shows processes in this two-step method.

This two-step method can be described in parallel pipeline in 3(b) using all KLT, AKLT and quadratic QKLT.

1. Compute optical flow $\boldsymbol{u}_K$, $\boldsymbol{u}_A$ and $\boldsymbol{u}_Q$ by KLT, AKLT and QKLT, respectively, for both stereo images and the displacement d_A by AKLT.



(a)



(b)

Figure 3. Two methods for scene flow computation (a) Two-step method for scene flow field computation. In the first step, the region on which the norms of optical flow vectors are sufficiently large is selected. In the second step, the scene flow vectors on the region selected in the first step are computed. (b) Pipeline for scene flow field computation. The method simultaneously computes all KLT methods with locally constant, affine and quadratic constrains.

2. Accept $\boldsymbol{u}_Q$ and d_A on points $|\boldsymbol{u}_K| \geq \epsilon$.

3. Compute $\dot{\boldsymbol{X}}$ using $\boldsymbol{u}_Q$ and d_A.

Figure 3(b) shows the pipeline of this two-phase method.

The spherical grids for spherical histogram are generated from an icosahedron. Form the original 20 faces, 96 faces are generated in our discretisation. Let $\{\boldsymbol{v}_i\}_{i=1}^{96}$ to be the centroid of each face. If we have the relation,

$$arg\min(\boldsymbol{v}_i^\top \frac{\dot{\boldsymbol{X}}}{|\dot{\boldsymbol{X}}|}) = \boldsymbol{v}_k \tag{38}$$

for the scene flow vector $\dot{\boldsymbol{X}}$, the norm $|\dot{\boldsymbol{X}}|$ is voted to bin F_k, whose centroid is $\boldsymbol{v}_k$.

Table 1. Examples of environments in front of a driving car

Driving status of autonomous car												
status of car	stopping	accelerating		constant		backing		turning left		turning right	
oncoming car		◯	×	◯	×	◯	×	◯	×	◯	×

relations a car in front			
inter-vehicle distance	stationary	increasing	decreasing

Over-takes and passing conditions			
passing condition	next lane for passing	in the same lane	
oncoming car		without	with

5. Numerical Experiments

Table 3 shows the first frames of the image sequences and the symbol strings yielded by our algorithm. These sequences are from KITTI sceneflowDataset2015. Table 2 shows the driving conditions of a car and the environments around the car. Figure 4 shows examples for the first frame of images and temporal transportations of the scene flow fields for 15 image sequences. In graphs in right column of Figure 4, blue and red lines illustrate the trajectory of the Wasserstein distance and linear approximation, respectively. Furthermore, for comparison, the first and second derivatives of the temporal transportations computed as temporal trajectory of the Wasserstein distance of scene flow fields between successive pairs of frames.

There is no sequence for over-taking and passing using the next lane. 15 results with real image sequences yield symbol strings with the same properties as those of synthetic image sequence. These results imply that the on-line algorithm detects the temporal transportation of scene flow fields as symbol strings while a car with a mounted camera is driving on streets in cities surrounded by buildings, obstacles and cars and suburban areas surrounded by buildings, trees and obstacles.

The results of our experiments clarified following properties.

1. Even if the scene flow vectors contain computational errors and noise, the time trajectory of the Wasserstein distance describes the temporal changes in the dominant direction of the scene flow fields.

2. The single peak which appears on the profiles of time trajectory of the Wasserstein distance corresponds to the change in the main driving direction while a car is driving on a curved street.

3. Our method produces the same symbol string to constant foward motion for the case with and without oncoming car in the next lane.

4. Property 3 implies that both the symbols and the scene flow field are required for the detection of a moving car in the next lane, when a car is moving forward with constant speed.

Experiments using real image sequences show that our motion symbolisation algorithms yield symbol strings, which correspond to the motion state of driving cars in various environments.

The combination of the street geometry in digital maps and driving conditions allow an on-board machine in a car to generate symbol strings of the motion situation. These inferred symbol strings are stored in a dictionary. If images captured by a car-mounted camera yield a symbol string with perturbations to entries in the dictionary, the machine infers abnormalities in the environment around the car.

The machine controls the car to avoid incidents corresponding to detected abnormalities using symbol strings. This inference of abnormalities by the on-board machine is achieved by string matching. Therefore, the symbolisation of temporal scene flow fields is suitable as a procedure for dictionary generation and for inference by entries of the dictionary, using string-matching algorithms.

6. Conclusions

In this paper, we have proposed a method for the symbolisation of temporal changes of the environments around the autonomous robot in work space using the scene flow field for recognition of events. Our method extracts words of motion in front of the robots from images captured by a robot-mounted stereo camera system.

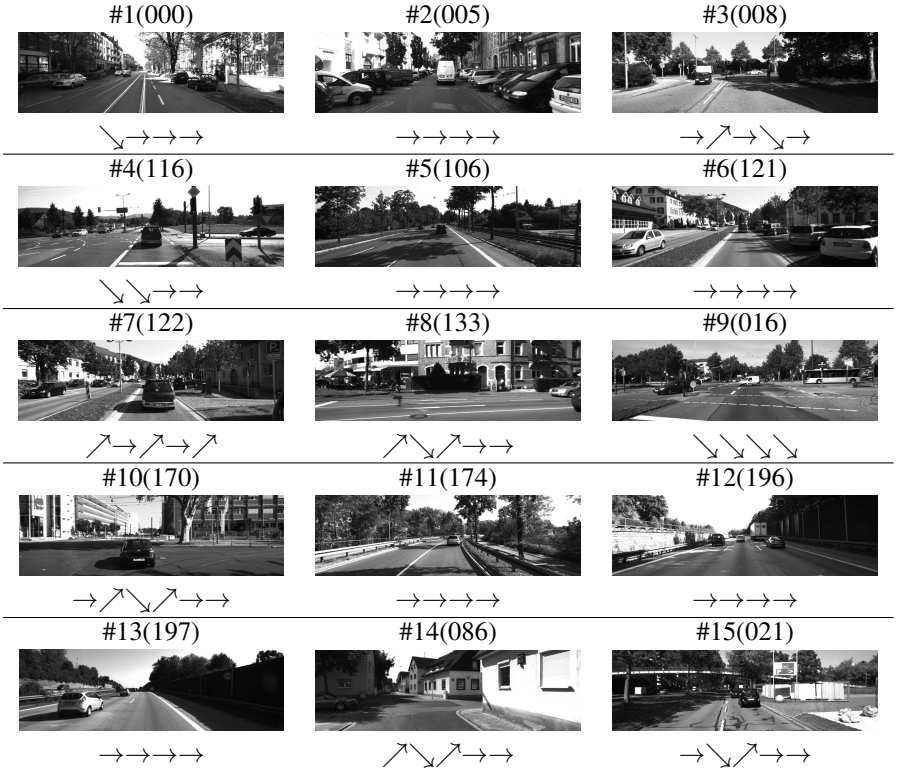
We have also extended the KLT tracker to use piecewise-linear and locally quadratic constrains for optical flow using a model-fitting scheme. For scene flow computation, we have used all the original KLT and our extensions. The KLT was used for the segmentation of moving region from background. The KLT with piecewise-linear and locally quadratic constrains were used for the computation of stereo disparities between stereo pair images and optical flow between frames in each image sequence.

This research was supported by the Grants-in-Aid for Scientific Research funded by the Japan Society for the Promotion of Science.

Table 2. Statuses of 15 sequences from KITTI sceneflowDataset2015.

No.	motion of car	front car	inter-vehicle dist.	oncoming car	additional conditions
#1	straight	without		with	shadow on the lane
# 2	straight	with	constant	without	parking in both sides
#3	straight	without		two cars	approaching to a cross
#4	straight	with	increasing	without	approaching to a cross
#5	straight	with	constant	with	
#6	straight	with	constant	with	with the centre lane
#7	accelerating	with	increasing	without	
#8	turning right	with	outside of screen	with in the first frame	
#9	stop after braking	with		without	approaching to crossing
#10	turning left	with	constant	without	
#11	right curve	with	constant	without	
#12	over-taking	with		without	the left car is passing the right car
#13	straight after over-taking	with		without	
#14	turning left after braking	without		with	truing a crossing in city
#15	changing lanes	with	increasing	without	shadow on lanes

Table 3. Symbols of 15 sequences from KITTI sceneflowDataset2015. Images are the first frames of sequences.

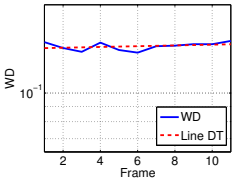


References

[1] Fisher, N. I., *Statistical Analysis of Circular Data*, Cambridge University Press, 1993.

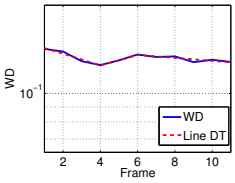
[2] Chaudhry, R., Ravichandran, A., Hager, G. D., Vidal, R., Histograms of oriented optical flow and binet-cauchy kernels

(a) # 1 image



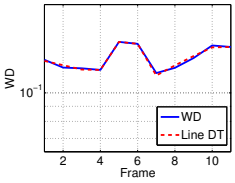
(b) # 1 linear transportation

(c) # 2 image



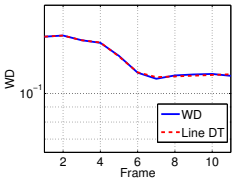
(d) # 2 linear transportation

(e) # 3 image



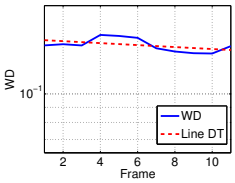
(f) # 3 linear transportation

(g) # 4 image



(h) # 4 linear transportation

(i) # 5 image



(j) # 5 linear transportation

Figure 4. Results of KITTIsceneflow2015 for sequences 000, 005, 008, 116 and 106.The blue and red lines illustrate the trajectory of the Wasserstein distance and linear approximation, respectively.

on nonlinear dynamical systems for the recognition of human actions, Proceedings of CVPR2009, 1932-1939, 2009.

[3] Rabin, J., Delon, J., Gousseau, Y., Transportation distances on the circle, JMIV, **41**, 147-167, 2011.

[4] Villani, C., *Optimal Transport, Old and New*, Springer, 2009.

[5] Wedel, A., Cremers, D., *Stereo Scene Flow for 3D Motion Analysis*, Springer, 2011.

[6] Vogel, Ch., Schindler, K., Roth, S., 3D scene flow estimation with a rigid motion prior, Proceedings of ICCV2011, 1291-1298, 2011.

[7] Vogel, Ch., Schindler, K., Roth, S., Piecewise rigid scene flow, Proceedings of ICCV2013, 1377-1384, 2013.

[8] Lucas, B. D., Kanade, T., An iterative image registration technique with an application to stereo vision, Proceedings of IJCAI1981, 674-679, 1981.

[9] Shi, J., Tomasi, C., Good features to track, Proceedings of CVPR1994, 593-600, 1994.

[10] Rubner, Y., Tomasi, C., Guibas, L. J., A metric for distributions with applications to image databases, Proceedings of ICCV1998, 59-66, 1998.

[11] Hwang, S.-H., Lee, U.-K., A hierarchical optical flow estimation algorithm based on the interlevel motion smoothness constraint, Pattern Recognition, **26**, 939-952, 1993.

[12] Amiaz, T., Lubetzky, E., Kiryati, N., Coarse to over-fine optical flow estimation, Pattern Recognition, **40**, 2496-2503, 2007.

[13] Hershberger. J., Snoeyink, J., Speeding up the Douglas-Peucker line-simplification algorithm, Proc 5th Symp. on Data Handling, 134-143, 1992.
UBC Tech Report TR-92-07 http://www.cs.ubc.ca/cgi-bin/tr/1992/TR-92-07