Topological Labelling of Scene using Background/Foreground Separation and Epipolar Geometry

Hiroki Hiraoka
School of Science and Engineering, Chiba University
Yayoi-cho 1-33, Inage-ku,
Chiba 263-8522, Japan

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

Abstract

The robust Principal Component Analysis (rPCA) efficiently separates an image into the foreground and background regions. The stixels provide middle-level expression of a scene using vertical columnar-superpixels of pixels with same depth computed from a pair of stereo image. Combining the classification of pixels by rPCA and depth map, topological labelling of pixels of each frame in an image sequence is achieved. The algorithm constructs static stixels and moving boxes of an image sequence from background and foreground regions, respectively. The algorithm also estimates free-space for motion planning from background regions as a collection of horizontal columnar-superpixels parallel to the epipolar lines.

1. Introduction

In this paper, using background/foreground separation based on robust principal component analysis (rPCA) [1, 2, 3, 4] and epipolar geometry [5], topological labelling of scenes is derived. The algorithm constructs static superpixels [6, 7], which lie on the static background, and moving superpixels, which lie on the moving regions. The algorithm also extracts moving obstacles [8, 9, 10] as moving boxes in an image sequence.

Stixels [11, 12], which are vertical columnar-superpixels computed from depth information, provides a middle-level visual-expression of scenes [13, 14]. Collection of stixels describes both static and dynamic obstacle regions [11, 12], which are corresponding to moving objects in the foreground and geometrical obstacles in the background, respectively. For reliable expression of scenes for autonomous navigation and driving, additional information, such as local geometry [15], semantics in the scene [16] derived by segmentation of scene, lider-based spatial observation [17], of each point and segment in a scene are demanded. Combination of background/foreground and depth information derives topological structure in a view [18, 19, 20] form a pair of stereo images.

Dealing with an image array as a matrix, rPCA extracts the foreground and background which correspond to sparse and low rank parts of an image array [1, 2, 3, 4, 10], respectively. In image sequences observed by a car-mounted imaging system, the background corresponds to the road and obstacles. Although stixels describe both dynamic and static obstacles in the front view, a free-space for motion planning in the front view is estimated using the geometric assumptions [20]. For autonomous driving and navigation, understanding of topological structure of scene is the next step of semantic labelling.

By separating each image in a sequence of images into the foreground and background images, the algorithm constructs stixels as vertical superpixels from background. From the compliment of stixel regions, the algorithm extracts a collection of horizontal columnar-superpixels, which are perpendicular to stixels on an image plane. The region occupied by these horizontal columnar-superpixels on the ground-plane regions correspond to free motion planning region. We call horizontal columnar-superpixels fixels. Moreover, applying depth map construction the foreground, we extract regions corresponding to moving obstacles. These procedures separate regions and affix labels to them based on topology and dynamics in scenes.

2. Topological Segmentation of Scene

2.1. Background/Foreground Separation using rPCA

The robust principal component analysis (rPCA) decomposes a large matrix array into a low-rank part and a sparse part. Measuring the sparsity of matrix $\boldsymbol{F}$ by $|\boldsymbol{F}|_0$, which is the cardinality of the non-zero elements of the matrix $\boldsymbol{F}$, minimisation of

$$J(\boldsymbol{L},\boldsymbol{S}) = rank(\boldsymbol{L}) + \lambda|\boldsymbol{S}|_0, \quad \boldsymbol{F} = \boldsymbol{L}+\boldsymbol{S}. \tag{1}$$

decomposes the $m \times n$ data matrix $\boldsymbol{F}$ into the matrices $\boldsymbol{L}$ and $\boldsymbol{S}$ with the conditions

$$rank(\boldsymbol{L}) \leq k \ll rank(\boldsymbol{F}), \ \ |\boldsymbol{S}|_0 \leq l \ll m \times n. \tag{2}$$

Equations (1) and (1) decompose matrices by balancing the relation between the dimensionality and scale of the decomposed data, assuming that the simplest and most useful data all lie near some low-dimensional subspace.

The minimisation problem of eq. (1) is relaxed to the minimisation problem

$$J_r(\boldsymbol{L},\boldsymbol{S}) = |\boldsymbol{L}|_* + \lambda|\boldsymbol{S}|_1, \quad \boldsymbol{F} = \boldsymbol{L}+\boldsymbol{S}, \tag{3}$$

where $|\boldsymbol{A}|_*$ and $|\boldsymbol{A}|_1$ are the nuclear norm and ℓ_1 norm of matrix $\boldsymbol{A}$, respectively, by replacing the minimisations of the rank and ℓ_0 norm of matrices to minimisation of the nuclear and ℓ_1 norms, respectively. This relaxed minimisation is achieved by minimising the following augmented Lagrangian,

$$\begin{aligned} J(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Y}) &= (|\boldsymbol{L}|_* + \lambda|\boldsymbol{S}|_1) + \langle \boldsymbol{Y}, \boldsymbol{F}-(\boldsymbol{L}+\boldsymbol{S})\rangle \\ &\quad + \frac{\mu}{2}|\boldsymbol{F}-(\boldsymbol{L}+\boldsymbol{S})|_F^2. \end{aligned} \tag{4}$$

Minimisation of eq. (4) is achieved by alternative minimisation of $\boldsymbol{L}$ and $\boldsymbol{S}$. Furthermore, the sparsity condition is preserved by applying thresholding to the elements of $\boldsymbol{S}$ in each iteration step. Moreover, the rank condition is preserved by applying singular value decomposition (SVD) to matrix $(\boldsymbol{F}-\boldsymbol{S})$ in each iteration. Therefore, setting the proximal operation and SVD operation as

$$\mathcal{S}_\tau(x) = \mathrm{sgn(x)}\max(|x|-\tau, 0) \tag{5}$$

for a scalar $x > 0$ and

$$\mathcal{D}_\tau(\boldsymbol{F}) = \boldsymbol{U}\boldsymbol{S}_\tau(\boldsymbol{\Sigma})\boldsymbol{V}^\top \tag{6}$$

for a matrix $\boldsymbol{F}$, this minimisation is decomposed to the alternate minimisations

$$\arg\min_{\boldsymbol{L}} J(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Y}) = \mathcal{S}_{\frac{\lambda}{\mu}}(\boldsymbol{F}-\boldsymbol{L}+\frac{1}{\mu}\boldsymbol{Y}), \tag{7}$$

$$\arg\min_{\boldsymbol{S}} J(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Y}) = \mathcal{D}_{\frac{1}{\mu}}(\boldsymbol{F}-\boldsymbol{S}+\frac{1}{\mu}\boldsymbol{Y}). \tag{8}$$

In eq. (7), the proximal operation is independently applied to each element of $\boldsymbol{F}-\boldsymbol{L}+\mu^{-1}\boldsymbol{Y}$.

Figure 1 shows the results of rPCA applied to the image illustrated on the left. We set $\mu = m \times n/4|\boldsymbol{F}|_1$ and $\lambda = 1/\sqrt{m \times n}$ for the $m \times n$ image array $\boldsymbol{F}$. Figure 1 shows that rPCA separates the backgrounds as low-rank image matrices from the original images.

2.2. Topological Properties of Segments

We decompose an $m \times n$ image array $\boldsymbol{F}$ into three image arrays such that

$$\boldsymbol{F} = \boldsymbol{M} + \boldsymbol{H} + \boldsymbol{T}, \tag{9}$$

where $\boldsymbol{M}$, $\boldsymbol{H}$ and $\boldsymbol{T}$ correspond to the moving obstacles in the foreground, the free-space on which objects in $\boldsymbol{M}$ move and static obstacles in the background. We derive matrix factorisation of eq. (9) from the minimiser of eq. (1).

For the derivation of $\boldsymbol{M}$, $\boldsymbol{H}$ and $\boldsymbol{T}$ from $\boldsymbol{L}$ and $\boldsymbol{S}$ using geometric properties of these matrices, we define matrix operations.

For the image $\boldsymbol{F} = ((f_i j))$, we define the lower array as

$$\boldsymbol{F}(u) = ((f_{ij}^u)), \quad f_{ij}^u = \begin{cases} f_{ij}, & \text{if } f_{ij} \leq u, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

The binalisation and projection of a matrix array are

$$\chi(\boldsymbol{F}) = ((\overline{f}_{ij})), \quad \overline{f}_{ij} = \begin{cases} 1, & \text{if } f_{ij} \neq 0 \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

and

$$\Pi_\Omega(\boldsymbol{F}) = ((\check{f}_{ij})), \quad \check{f}_{ij} = \begin{cases} f_{ij}, & \text{if } (i,j) \in \Omega, \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

for an appropriate rectangle set $\Omega \subset [1,m] \times [1,n]$, respectively. The inclusion relation of matrices is defined as

$$\chi(\boldsymbol{F}) \sqsubset \chi(\boldsymbol{G}) \Leftrightarrow \{\overline{f}_{ij} \neq 0\}_{i=1,j=1}^{m,n} \subset \{\overline{g}_{ij} \neq 0\}_{i=1,j=1}^{m,n}. \tag{13}$$

For a triplet of $m \times n$ matrices $\boldsymbol{A} = ((a_{ij}))$, $\boldsymbol{B} = ((b_{ij}))$ and $\boldsymbol{C} = ((c_{ij}))$, we define

$$\boldsymbol{C} = \boldsymbol{A} \sqcup \boldsymbol{B} \Leftrightarrow \overline{c}_{ij} = \max(\overline{a}_{ij}, \overline{b}_{ij}). \tag{14}$$

Since $\boldsymbol{S}$ and $\boldsymbol{L}$ correspond to moving foreground, which occupy a collection of small regions, and background, which are static, respectively, we assume the relations

$$\begin{aligned} \boldsymbol{L} &= \boldsymbol{H} + \boldsymbol{T} + \boldsymbol{E}, && (15) \\ \boldsymbol{O} &= \chi(\boldsymbol{H}) \odot \chi(\boldsymbol{T}), && (16) \\ \chi(\boldsymbol{T}) &\sqsubset \chi(\boldsymbol{F} \odot \chi(\boldsymbol{L})), && (17) \\ \chi(\boldsymbol{M}) &\sqsupset \chi(\boldsymbol{F} \odot \chi(\boldsymbol{S})), && (18) \\ \chi(\boldsymbol{M}) &= \Pi_R \chi(\boldsymbol{M}) && (19) \end{aligned}$$

for

$$|\boldsymbol{E}|_0 < c, \quad |\Pi_R\chi(\boldsymbol{M}) - \chi(\boldsymbol{F} \odot \chi(\boldsymbol{S}))|_0 < \varepsilon, \tag{20}$$

where c and ε are a positive constant and a positive small constant, respectively, and $\boldsymbol{O}$ is the zero matrix. In eqs. (16), (17), (18) and (19) $\boldsymbol{A} \odot \boldsymbol{B} = ((a_{ij}b_{ij}))$ is the

(a) Original $\boldsymbol{F}$ (b) Low-rank $\boldsymbol{L}$ (c) Sparse $\boldsymbol{S}$

(d) Original $\boldsymbol{F}$ (e) Low-rank $\boldsymbol{L}$ (f) Sparse $\boldsymbol{S}$

(g) Original $\boldsymbol{F}$ (h) Low-rank $\boldsymbol{L}$ (i) Sparse $\boldsymbol{S}$

(j) Original $\boldsymbol{F}$ (k) Low-rank $\boldsymbol{L}$ (l) Sparse $\boldsymbol{S}$

Figure 1. Examples of rPCA applied to images from KITTI data set. Images in the left, middle and right columns are the original, low-rank and sparse image arrays, respectively. rPCA separates the backgrounds as low-rank image matrices from the original images.

Hadamard product of matrices $\boldsymbol{A} = ((a_{ij}))$ and $\boldsymbol{B} = ((b_{ij}))$.

Equations (15) and (16) imply that regions corresponding to $\boldsymbol{H}$ and $\boldsymbol{T}$ are mutually compliment in the background. Equation (17) implies that static stixels are contained in the low-rank part of the image matrix. Equations (18) and (19) imply that non-zero elements in $\boldsymbol{M}$ are a collection of moving boxes which contain $\boldsymbol{S}$. Equation (20) describes minimisation criteria on labelling of regions based on matrix decomposition.

If $\boldsymbol{S}$ is a collection of connected components which are not overlapping, $\boldsymbol{M}$ can be decomposed into $\{\boldsymbol{M}_k\}_{k=1}^{n}$ which satisfy the condition

$$\boldsymbol{M} = \bigsqcup_{k=1}^{n} \boldsymbol{M}_k, \ \ \chi(\boldsymbol{M}_k) = \Pi_{\Omega_k}(\chi(\boldsymbol{M}_k)) \qquad (21)$$

for

$$\Omega_k = [m_{k(1)} \leq i \leq m_{k(2)}] \times [n_{k(1)} \leq j \leq n_{k(2)}] \qquad (22)$$

in the image grid $[1, m] \times [1, n]$.

3. Matrix Decomposition using Epipolar Geometry

3.1. Pixel Labelling in the Background

Stixels are mathematically vertical columnar-superpixels with the same depth in the background of an image [14]. We extract static stixels from $\boldsymbol{F} \odot \chi(\boldsymbol{L})$ in a stereo pair. Furthermore, for smoothing of the background, the operation

$$B(\boldsymbol{L}^{(t)}) := B(\boldsymbol{L}^{(t+1)}) \odot (-2B(\boldsymbol{L}^{(t)})) \odot B(\boldsymbol{L}^{(t-1)}) \quad (23)$$

is applied, where $\boldsymbol{L}^{(t)}$ is the low-rank matrix part of the images at the frame t.

Since the rPCA separates an image into the foreground and background, we omit the free-space estimation step and height evaluation step from the standard stixel estimation procedure [12, 13, 14]. We only adopt depth information [12, 13] for the estimation of vertical columnar-superpixels as obstacle regions. The we construct image matrix $\boldsymbol{T}$ from pixels in the stixels.

3.2. Pixel Labelling using Epipolar Geometry

We define horizontal columnar-superpixels in the background of an image for the construction of the matrix $\boldsymbol{H}$.

Definition 1 *A fixel (Free space pIXEL) is a collection of*

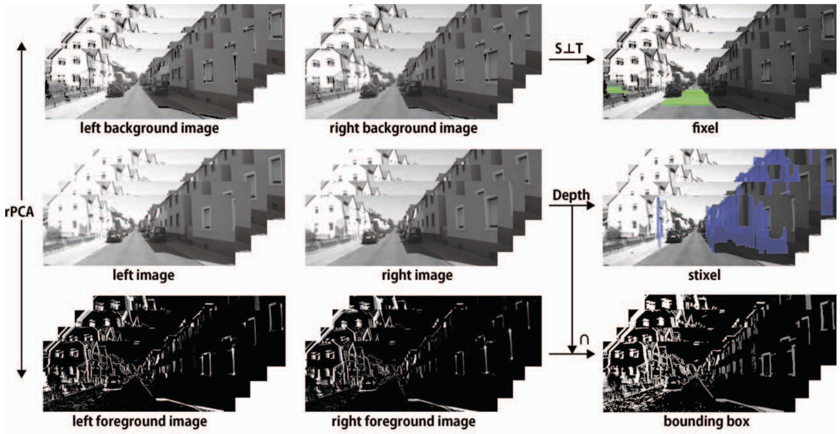


Figure 2. Pipeline for image decomposition and labelling. The algorithm first separates the background and foreground by applying rPCA to the image matrix. The algorithm then extracts the depth of each pixel and epipolar lines on an image from a stereo pair image in a stereo sequence. Then, the algorithm constructs stixels and removes stixel regions from the background. Second, the algorithm determines the fixels as horizontal columnar-superpixels using vanishing lines and epipolar lines. Finally from the foreground, the algorithm extracts the moving boxes with depth information.

pixels along an epipolar lines between a pair of vanishing lines on a planar area in the background.

This definition implies that fixels are horizontal columnar-superpixels on $\chi(\boldsymbol{H})$ with width one, which are perpendicular to stixels in $\boldsymbol{L}$.

Algorithm 1: Fixel construction

Input: time series image
 Compute rPCA for an image array.
 repeat
  Compute edge pixels.
  Compute stixels.
  Compute epipolar lines.
  Assign labels on pixels along epipolar lines
  Compute fixels
 until Compute free space for all frames
Output: a sequence of fixels

Algorithm 1 describes the procedure for the fixel construction to compute the image matrix $\boldsymbol{H}$ from the background computed by rPCA. First, the algorithm separates the background using rPCA for image matrix. The algorithm constructs stixels in the background and removes stixel regions from the background. Finally, using vanishing lines and epipolar lines on the image, the algorithm determines the fixels as horizontal columnar-superpixels in the complement region to the region occupied stixels.

Using local geometric properties of pixels along epipolar lines, we extract the fixels in the background. In this step, we remove thin and short stixels as shown in Figure 3. Figures 3(a) and 3(b) describe this removing step after removing thin stixels. This process guarantees stable extraction of fixels from the complement regions of the area occupied by stixels.

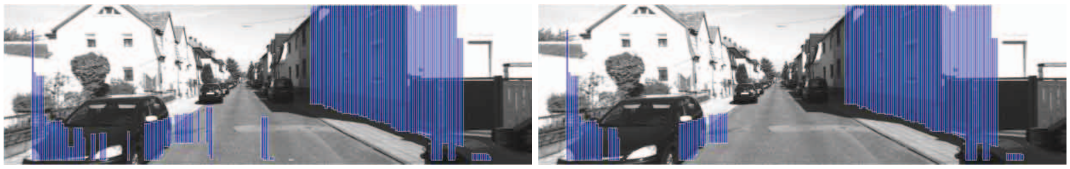
Assigning that symbols e, p and s are assigned to pixels on an edge line (a relaxed line of the vanishing line), planar area and stixels, respectively, the symbol sequence of pixels f along epipolar lines between a pair of vanishing lines on a plane is

$$f = *p^m * *p^n *, \tag{24}$$

where $* \in \{e, s, \phi\}$. From each labelled sequence, we extract the subsequences $*p^n*$ that are longer than a predefined threshold from each f. Since the slope of a horizontal epipolar line is small, we assign labels for a straight strip line whose slope is small. Assuming S and W to be the slope and width of the strip, respectively, we compute n such that $S \times 10^n > 1$ subject to $10^n < W$.

Figure 4 illustrates configurations of an epipolar line and obstacles on an image plane. Figure 4(a) shows a string of labels on an epipolar line and obstacles without any car in the front view. Figure 4(b) shows a string of labels on an epipolar line and obstacles with a car, which separates pixels along the epipolar line into two parts, in the front view.

Figures 5(a) and 5(b) describe state transitions of a symbol sequence on an epipolar line. The automaton in Fig. 5(a) accepts ideal sequences without noise caused by extracted edges on the road. The automaton in Fig. 5(b) accepts sequences with noise caused by extracted edges on the road.



(a) (b)

Figure 3. Stixel selection. Thin and short stixels are removed before the fixel-labelling procedure. (a) and (b) show the image with thin and short stixels and the image after removing thin and short stixels.

(a) (b)

Figure 4. Configuration of labels on an epipolar line. (a) String of labels on an epipolar line and obstacles without any car in the front view. (b) String of labels on an epipolar line and obstacles with a car, which separates pixels along an epipolar line into two parts, in the front view. Symbols e, p and s are assigned to pixels on an edge line (a relaxed line of the vanishing line), planar area and stixels, from each labelled sequence, we extract the maximum sequence of p^n* from each f.

3.3. Moving Box Separation using Depth Map

Depth information in $\boldsymbol{F} \odot \chi(\boldsymbol{S})$ computed from stereo pairs decomposes $\boldsymbol{M}$ into $\{\boldsymbol{M}_k\}_{k=1}^n$. We call each $\boldsymbol{M}_i$ a moving box. Appling depth reconstruction for filtered stereo pairs

$$\overline{\boldsymbol{F}_+} = \boldsymbol{F}_+ \odot \Pi_{R_+}(B(\boldsymbol{S}_+)), \tag{25}$$

$$\overline{\boldsymbol{F}_=} = \boldsymbol{F}_- \odot \Pi_{R_-}(B(\boldsymbol{S}_-)), \tag{26}$$

where $+$ and $-$ express the left and right images, respectively, we compute three dimensional configurations of portions in the foreground in an image.

Setting $\boldsymbol{D}$ to be the depth map computed from $\overline{\boldsymbol{F}_+}$ and $\overline{\boldsymbol{F}_+}$, we reset

$$\boldsymbol{M} = \boldsymbol{N} \odot \chi(\boldsymbol{D}(u)), \quad \boldsymbol{N} = \boldsymbol{F} \odot \chi(\boldsymbol{S}), \tag{27}$$

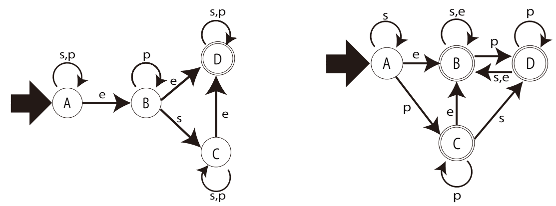
assuming that depths of pixels in the moving foreground is shallow. On $\boldsymbol{M}$, we define the moving boxes for disconnected portions in $\boldsymbol{S}$. Then, we accept the moving boxes larger than a predetermined size. Classification of pixels using depth decomposes $\boldsymbol{M}$ into $\{\boldsymbol{M}_i\}_{i=1}^m$ in each box. After detecting box region and depth of each pixel in each box, the description of topology of boxes is achieved by Algorithm 2. These boxes might be overlapping, if parts of object with deep depth are occluded by objects in near sides.

Definition 2 *Let v and h be the lengths of the vertical and horizontal edges of rectangles, respectively. For $\gamma = h/v$, we call horizontal and vertical rectangles, if $\gamma > 1$ and $\gamma < 1$, respectively.*

Algorithm 2: Dynamic box extraction

Input: time series image
  Compute moving box for a frame
  repeat
    Set boxes
    Compute the depth of pixels
    Construct depth histogram
    Classify pixels using depths
    Affix depth labels
    Construct boxes with depths
  until moving boxes for all frames
Output: a sequence of boxes

The rPCA-based foreground/background separation extract artefacts caused by shadows on the road as a sparse part of an image array matrix. A region separated from shadows yields a collection boxes, whose horizontal edges are longer than vertical edges, encircled by fixel areas. Therefore, we remove boxes if these boxes are horizontal rectangles encircled by fixel areas.



(a) (b)

Figure 5. Automata which accept sequence p^n from the labelled sequence on an epipolar line. (a) Automaton for ideal sequences without noise caused by extracted edges on the road. (b) Automaton for practical ideal sequences with noise caused by extracted edges on the road.

4. Numerical Examples

Figure 6 shows the results for real KITTI image sequence. In these two examples, (a) and (c) are the left and right images of a stereo sequence. (b) is the left image with epipolar lines. (d) and (f) are the left and right images of the stixels. (e) is the left image of the fixels.

In Figure 7, each figure shows results on image decomposition. In each image of these figures, left two images in the middle row are input stereo images with rectification. Furthermore, images in the right column, From top to bottom, fixel, stixel and moving box images, respectively. In these examples, the horizontal rectangles with the condition $\gamma \geq 2.5$ are removed.

For the removal of motion artefacts caused by shadows in RGB-colour images sequences, selection of the colour space is the fundamental problem [21, 22], since removal and detection of artefacts caused by shadows depends on the colour-space selection. In our experiment, geometric assumption removes artefacts caused by shadows from monochrome image sequences.

Figure 8 shows the heat map of depths of moving boxes. From top to bottom, frames 199, 200, 201 and 202 of sequence # 0015 of KITTI dataset are shown. From shallow to deep for depth, colour changes from red to blue. These sequences show the motion of car using boxes and depth colour labels.

These two results demonstrate the performance of our method on topological labelling from foreground/background images and the depth map.

5. Conclusions

For autonomous driving, estimation of topological structure and detection of moving obstacles in the front view are essential tasks for safe driving.

Our algorithm extracts vertical and horizontal columnar-superpixels as stixels and fixels, respectively and moving box in the front view. By separating each image in a sequence into the background and foreground, the algorithm constructs stixels from background as vertical columnar-superpixels. From the complement of stixel regions in background, the algorithm extracts a collection of horizontal columnar-superpixels, which are called fixels. Morover, we extract moving obstacles with depth information from the foreground. Stixels and moving boxes describe static obstacles and dynamic objects in the font view. The fixels on the ground-plane regions correspond to free space for motion planning.

References

[1] Emmanuel J. Candès, Xiaodong Li, Yi Ma, John Wright: Robust principal component analysis? J. ACM, **58**, 1:11-1:37, 2011.

[2] John Wright, Arvind Ganesh, Shankar Rao, Yigang Peng, Yi Ma: Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization, Advances in Neural Information Processing Systems 22 (NIPS 2009) 2080-2088, 2009.

[3] Thierry Bouwmans, El-Hadi Zahzah: Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance, CVIU, **122**, 22-34, 2014.

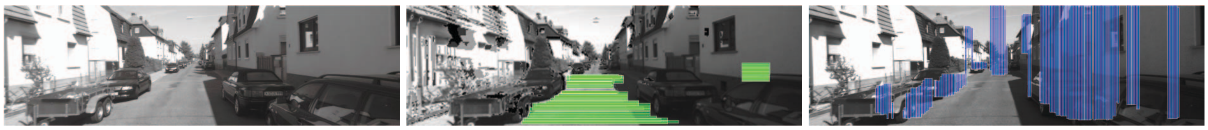
[4] Sajid Javed, Arif Mahmood, Thierry Bouwmans, Soon Ki Jung: Background-foreground modeling based on spatiotemporal sparse subspace clustering, IEEE TIP, **26**, 5840-5854, 2017.

[5] Kenichi Kanatani: *Geometric Computation for Machine Vision (Oxford Engineering Science Series)* OUP, 1993.

[6] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, Sabine Süsstrunk: SLIC superpixels compared to state-of-the-art superpixel methods, IEEE PAMI, **34**, 2274-2282, 2012.

[7] Murong Wang, Xiabi Liu, Yixuan Gao, Xiao Ma, Nouman Q. Soomro: Superpixel segmentation: A benchmark, Signal Processing: Image Communication **56**, 28-39, 2017.

(a) input(left) (b) epipolar line (c) input(right)



(d) stixel(left) (e) fixel (f) stixel(right)

Figure 6. Example for real images. (a) and (c) are the left and right images, respectively, of a stereo sequence. (b) is the left image with epipolar lines. (d) and (f) are the left and right images, respectively, of the stixels. (e) is the left image of the fixels.

[8] Hongwei Yong, Deyu Meng, Wangmeng Zuo, Lei Zhang: Robust online matrix factorization for dynamic background subtraction, arXiv:1705.10000v1, 2017.

[9] N. Benjamin Erichson, Carl Donovan: Randomized low-rank dynamic mode decomposition for motion detection, CVIU, **146**, 40-50, 2016.

[10] N. Benjamin Erichson, Steven L. Brunton, J. Nathan Kutz: Compressed dynamic mode decomposition for background modeling, J. Real-Time Image Proc., https://doi.org/10.1007/s11554-016-0655-2, 2016.

[11] Ernst Dieter Dickmann: *Dynamic Vision for Perception and Control of Motion*, Springer, 2007

[12] Marius Cordts, Timo Rehfeld, Lukas Schneider, David Pfeiffer, Markus Enzweiler, Stefan Roth, Marc Pollefeys, Uwe Franke: The stixel world: A medium-level representation of traffic scenes, Image and Vision Computing, **68**, 40-52, 2017.

[13] Hernán Badino, Uwe Franke, David Pfeiffer: The stixel world - A compact medium level representation of the 3D-world, LNCS, **5748**, 31st DAGM(Pattern Recognition), 51-60, 2009.

[14] David Pfeiffer, Uwe Franke: Efficient representation of traffic scenes by means of dynamic stixels, IEEE 2010 Intelligent Vehicles Symposium, 218-224, 2010.

[15] Daniel Hernández Juárez,*et. al.*: Slanted stixels: representing San Francisco's steepest wtreets, BMVC 2017.

[16] Lukas Schneider *et. al*: Semantic stixels:depth is not enough, 2016 IEEE Intelligent Vehicles Symposium,110-114, 2016.

[17] Noor Haitham Saleem, *et. al*: Monocular stixels: A LIDAR-guided approach, 2018IVCNZ, 2018.

[18] Naoya Ohnishi, Atsushi Imiya: Featureless robot navigation using optical flow. Connect. Sci. 17, 23-46, 2005.

[19] Naoya Ohnishi, Atsushi Imiya: Appearance-based navigation and homing for autonomous mobile robot. Image Vision Comput. 31, 511-532, 2013

[20] Naoya Ohnishi, Atsushi Imiya: Dominant plane detection from optical flow for robot navigation. Pattern Recognition Letters **27**, 1009-1021, 2006.

[21] Yana Mileva, Andrés Bruhn, Joachim Weickert: Illumination-robust variational optical flow with photometric invariants, LNCS, **4713** (DAGM 2007), 152-162, 2007

[22] Ming-Ying Fan, Atsushi Imiya, Kazuhiko Kawamoto, Tomoya Sakai: Affine colour optical flow computation, LNCS, **8047** (CAIP2013), 507-514, 2013.

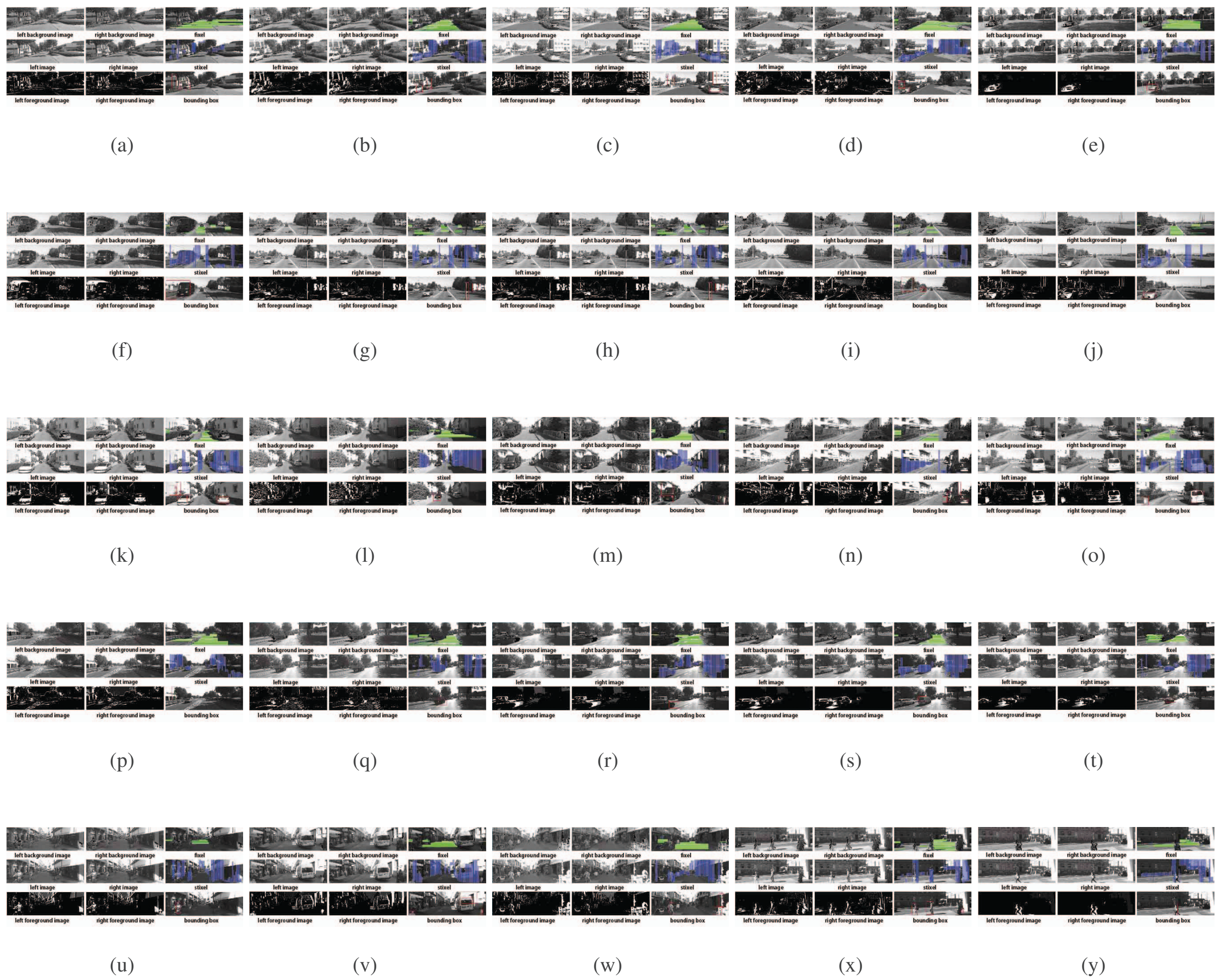


Figure 7. Decomposition Examples. From top to bottom results for sequences 0009, 0015, 0039, 0051 and 0071 are shown. For sequence 0009, from left to right results in frames 027, 035, 225, 271 and 413 are shown. For sequence 0015, from left to right results in frames 116, 190, 200, 215 and 242 are shown. For sequence 0039, from left to right results in frames 169, 187, 294, 314 and 363 are shown. For sequence 0051, from left to right results in frames 023, 186, 230, 303 and 327 are shown. For sequence 0071, from left to right results in frames 036, 122, 163, 970 and 1027 are shown.

(a) (b) (c) (d)

Figure 8. Depth map of moving box. From top to bottom, frames 199, 200, 201 and 202 of sequence # 0015 of KITTI dataset are shown. In this sequence from shallow to deep for depth, colour labels change from red to blue.