

Description

Method for image segmentation matching

5 Technical Field

The invention relates to a computer implemented method for image segmentation matching in the fields of computer visions and deep learning for autonomous driving. The invention further relates to a processing circuitry configured to run a computer
10 program for image segmentation matching, and a computer readable medium on which a computer program for image segmentation matching is stored.

Background

15 As a critical and fundamental technique in vision-based artificial systems, image matching is widely used in many applications, such as object detection, image recognition, image retrieval, vehicle re-identification, place recognition, accurate localization and Simultaneous Localization And Mapping (SLAM). The target of matching task is to solve similarity correspondence problem for contents from image
20 pairs. In traditional image matching methods, the handcrafted local features from the statistics of pixels or gradient information are used. The similarity of features pairs is commonly computed using different predefined metrics, like L2 distance and Cosine distance. More efficient neighborhood information for the statistics may be obtained by computing circular patterns with an adjustable radius. However, these
25 handcraft features are not robust enough to the changes of different viewpoints, illuminations and transformations. Consequently, the matching performance for the methods based on handcraft local features is not stable.

Summary of the invention

30

There may be a desire to provide an improved method for image segmentation matching.

The problem is solved by the subject-matter of the independent claims.

Embodiments are provided by the dependent claims, the following description and the accompanying figures.

- 5 The described embodiments similarly pertain to the computer implemented method, the processing circuitry, and the computer readable medium. Synergetic effects may arise from different combinations of the embodiments although they might not be described in detail.
- 10 Further on, it shall be noted that all embodiments of the present invention concerning a method might be carried out with the order of the steps as described, nevertheless this has not to be the only and essential order of the steps of the method. The herein presented methods can be carried out with another order of the disclosed steps without departing from the respective method embodiment, unless
- 15 explicitly mentioned to the contrary hereinafter.

Technical terms are used by their common sense. If a specific meaning is conveyed to certain terms, definitions of terms will be given in the following in the context of which the terms are used.

20

- According to a first aspect, a computer implemented method for image segmentation matching of a first segmentation y_k of a first image with a second segmentation x_k of a second image, where the first segmentation y_k and second segmentation x_k are a pair of segmentations, is provided. The method comprises the
- 25 steps learning joint features and metric of the first segmentation y_k and the second segmentation x_k using joint feature and metric learning, and regulating the joint feature and metric learning using a graph G^{y_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and neighboring segmentations of the first segmentation y_k , and using the second
- 30 segmentation x_k .

The segmentations x_k and y_k are a k -th pair of two images, comprising, for example, the same object or part of an object, therefore x_k corresponds to y_k in this regard. For the regulation, the segmentation x_k of the second image is used and the segmentations of the neighbor segmentations of the corresponding y_k segmentation are used. That is, the neighbor segmentations specified by G^{y_k} are segmentations of the first image. These neighbor segmentations of the first image and the segmentation x_k of the second image are input to a regulation network. The first image is also referred to as frame F_y , and the second image also referred to as frame F_x in this disclosure. The k -th pair of segmentations (x_k, y_k) may be a randomly sampled pair.

For the regulating of the joint feature and metric learning, additionally a graph G^{x_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and a second segmentation x_k may be used.

According to an embodiment, the method is executed within a framework, which comprises a basic model for the joint feature and metric learning, a graph-based regularization part for regulating the joint feature and metric learning, and a loss layer. The basic model comprises a first image feature extraction module, a second image feature extraction module and a fully connected neural network. The method further comprises the following steps:

Extracting first image features $f(y_k)$ of the first segmentation y_k by the first image feature extraction module and outputting the extracted first image features $f(y_k)$ to the fully connected neural network.

Extracting second image features $f(x_k)$ of the second segmentation x_k by the second image feature extraction module and outputting the extracted second image features $f(x_k)$ to the fully connected neural network.

Comparing the extracted first image features $f(y_k)$ and the extracted second image $f(x_k)$ features and outputting a predicted result $\hat{\alpha}_k$ to the loss layer (106), by the fully connected neural network, and

calculating a total loss l_{total} as a sum of the loss l_{ce} based on the predicted result $\hat{\alpha}_k$ and a loss l_{reg} based on the result of the graph-based regularization part, by the loss layer.

- 5 According to an embodiment, the graph-based regularization part comprises a third image feature extraction module, a fourth image feature extraction module, a graph attention network, and a discriminator, and the method further comprises the following steps:
 Extracting third image features $f(x_k)$ of the second segmentation x_k by the third
 10 image feature extraction module and outputting the extracted third image features $f(x_k)$ to the discriminator.
 Extracting fourth image features $f(y_l)$ of neighboring segmentations contained in graph G^{y_k} of the first image segmentation y_k by the fourth image feature extraction module and outputting the extracted fourth image features $f(y_l)$ to the graph attention
 15 network.
 Executing attention functions over the fourth image features $f(y_l)$, in a multi-head module to obtain high dimensional features $g(G^{y_k})$ of the neighboring segments, and outputting the high dimensional features $g(G^{y_k})$ of the neighboring segmentations to the discriminator, by the graph attention network, and
 20 comparing the extracted third image features $f(x_k)$ with the high dimensional features $g(G^{y_k})$ of the neighboring segmentations contained in graph G^{y_k} in layers of the discriminator and outputting the result $L_{empirical-ID}(x, G^y)$ to the loss layer, by the discriminator.
- 25 According to an embodiment, additionally the above steps are performed with respect to image features $f(y_k)$ of the first segmentation y_k and image features $f(x_l)$ of neighboring segmentations contained in graph G^{x_k} of the second image segmentation x_k , and to compare third image features $f(y_k)$ of the first segmentation (y_k) with the high dimensional features $g(G^{x_k})$ of the neighboring segmentations in
 30 layers of the discriminator (108) and outputting the result $L_{empirical-ID}(y, G^x)$ to the loss layer, by the discriminator. These steps are:

Extracting third image features $f(y_k)$ of the second segmentation y_k by the third image feature extraction module and outputting the extracted third image features $f(y_k)$ to the discriminator.

Extracting fourth image features $f(x_l)$ of neighboring segmentations contained in graph G^{xk} of the first image segmentation x_k by the fourth image feature extraction module and outputting the extracted fourth image features $f(x_l)$ to the graph attention network.

Executing attention functions over the fourth image features $f(x_l)$, in a multi-head module to obtain high dimensional features $g(G^{xk})$ of the neighboring segments, and outputting the high dimensional features $g(G^{xk})$ of the neighboring segmentations to the discriminator, by the graph attention network, and

comparing the extracted third image features $f(y_k)$ with the high dimensional features $g(G^{xk})$ of the neighboring segmentations contained in graph G^{xk} in layers of the discriminator and outputting the result $L_{\text{empirical-ID}}(y, G^x)$ to the loss layer, by the discriminator.

According to an embodiment, the discriminator comprises a bilinear layer with a trainable matrix M .

According to an embodiment, the bilinear layer form is given as $d(a, b) = \sigma(a^T M b)$, where $\sigma(\cdot)$ denotes the sigmoid function, where, a^T is a weight vector, and a and b are placeholders for $f(x_k)$ or $f(y_k)$, wherein the discriminator computes a loss $L_{\text{pair-empirical-ID}}$ according to a discrimination function, which is defined as

$$L_{\text{pair-empirical-ID}} = L_{\text{empirical-ID}}(X, G^y) + L_{\text{empirical-ID}}(y, G^x) =$$

$$= \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{x_k \leftrightarrow y_k\}} \log[d(f(x_k), g(G^{y_k}))] + \mathbf{1}_{\{x_k \nleftrightarrow y_k\}} \log[1 - d(f(x_k), g(G^{y_k}))] \right\}$$

$$+ \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{y_k \leftrightarrow x_k\}} \log[d(f(y_k), g(G^{x_k}))] + \mathbf{1}_{\{y_k \nleftrightarrow x_k\}} \log[1 - d(f(y_k), g(G^{x_k}))] \right\},$$

wherein $\mathbf{1}\{-\}$ is an indicator and $x_k \leftrightarrow y_k$ or $x_k \nleftrightarrow y_k$ denotes a match or mismatch with respect to a k th segmentation pair (x_k, y_k) .

According to an embodiment, the loss l_{reg} is calculated as loss $l_{\text{reg}} = -L_{\text{pair-empirical-ID}}$, by the loss layer.

- 5 According to an embodiment, a loss function for calculating the loss l_{ce} is a cross entropic over pairs of segmentation samples using a ground-truth label and the predicted result from the basic model, and the result obtained from graph-based regularization part, wherein the total loss is the weighted sum of the losses l_{ce} and l_{reg} .

10

According to an embodiment, the loss function l_{ce} is defined by

$$\ell_{\text{ce}} = -\frac{1}{k} \sum_k \{ \alpha_k \log \hat{\alpha}_k + (1 - \alpha_k) \log(1 - \hat{\alpha}_k) \},$$

wherein k is a k -th pair of segmentation samples, α_k a ground-truth label (, i.e., “1” for matched or “0” for unmatched label), and $\hat{\alpha}_k$ is a predicted result from the basic
15 model based on a sigmoid function.

According to an embodiment, the first image contains a street scene from a first perspective under first environmental conditions and the second image contains a street scene from a second perspective under second environmental conditions.

20

According to an embodiment, the first and the second image segmentations comprise spatial information.

The spatial information could get from sensors like radar, LiDAR, stereo camera and
25 depth camera or even from other estimation methods like the monocular depth estimation networks.

25

According to an embodiment, the first, the second, the third, and the fourth image feature extraction modules are one shared network, which serves as feature
30 descriptor function f and extracts the first, second, third and fourth image features as high dimensional features from the respective image segmentations.

30

According to an embodiment, the shared network is a residual neural network, ResNet.

According to a further aspect, a computer readable medium is provided on which a computer program for image segmentation matching of a first segmentation of a first image with a second segmentation of a second image is stored. The computer program comprises a basic model module configured to learn joint features and metric of a first segmentation of a first image and a second segmentation of a second image, and a graph-based regularization part configured to regulate the joint feature and metric learning using a graph G^{y_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and neighboring segmentations of the first segmentation y_k ; and using the second segmentation x_k .

According to a further aspect, a processing circuitry configured to run a computer program for image segmentation matching of a first segmentation of a first image with a second segmentation of a second image is provided. The computer program comprises a basic model module configured to learn joint features and metric of a first segmentation of a first image and a second segmentation of a second image, and a graph-based regularization part configured to regulate the joint feature and metric learning using a graph G^{y_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and neighboring segmentations of the first segmentation y_k ; and using the second segmentation x_k .

The computer program may be part of a further computer program, but it can also be an entire program by itself. For example, the computer program may be used to update an already existing computer program to get to the present invention.

The processing circuit may comprise circuits without programmable logics or may be or comprise a processor, a signal processor, a micro controller, a field programmable gate array (FPGA), an ASIC, a Complex Programmable Logic

Devices (CPLD), and/or any other programmable logic devices known to person skilled in the art.

The computer readable medium may be seen as a storage medium, such as for example, a USB stick, a CD, a DVD, a data storage device, a hard disk, or any other medium on which a computer program as described above can be stored.

These and other features, aspects and advantages of the present invention will become better understood with reference to the accompanying figures and the following description.

Short Description of the Figures

Fig. 1 shows a block diagram of an image segmentation matching framework,

Fig. 2 shows a block diagram of an example of a Graph Attention Network (GAT) block,

Fig. 3 shows a block diagram of an example of the structure of the image feature extraction module,

Fig. 4 shows a block diagram of an example of the fully connected neural network,

Fig. 5 shows a block diagram of an example of the discriminator,

Fig. 6 shows a flow diagram of the method,

Fig. 7 shows a flow diagram of further steps of the method, in particular relating to the basic model,

Fig. 8 shows a flow diagram of further steps of the method, in particular relating to the regulating of the joint feature and metric learning.

Detailed Description of Embodiments

Corresponding parts are provided with the same reference symbols in all figures.

The notation x is used to denote a random sample from the image segmentations and F^x is used to denote the frame from which x is extracted. The corresponding probability space for x is denoted by $(\mathcal{S}_{F^x}, 2^{\mathcal{S}_{F^x}}, \mathbb{P}_{F^x})$, where $\mathcal{S}_x$ is the sample

space of all available segmentations from $\mathcal{F}^x$, $2^{\mathcal{S}_x}$ denotes the collection of all subsets of $\mathcal{S}_x$, which is also a σ -algebra, $\mathbb{P}_{\mathcal{F}^x}(\{x\}) = p_{\mathcal{F}^x}(x)$ with $p_{\mathcal{F}^x}(\cdot)$ being the probability mass function.

- 5 The notation G^x is used to denote the neighborhood graph with respect to x . G^x is constructed with vertices being the K image segmentations (objects) around x and edges being established within all pairs of the K vertices. In other words, G^x is a complete graph. $E(G^{x_k})$ (an all-ones matrix) is used to denote the adjacent matrices of G^{x_k} , which represent the connection relationships between vertices in the
- 10 corresponding neighborhood graphs. The probability space for random graphs is $(\mathcal{G}, 2^{\mathcal{G}}, \mathbb{P}_{\mathcal{G}^x})$, where $\mathcal{G}$ is the set of all graphs with vertex set $\mathcal{S}$.

- Given two arbitrary frames F^x and F^y , let (x_k, y_k) be the k -th pair of segmentations random sampled independently from $\mathbb{P}_x \times \mathbb{P}_y$. Let G^{x_k} and G^{y_k} denote the
- 15 neighborhood graphs around x_k and y_k respectively.

- Fig. 1 shows block diagram of an image segmentation matching framework 100. The framework 100 consists of two parts 102, 104, which are the basic model 102 for the joint feature and metric learning and the graph-based regularization part 104 for regulating the joint feature and metric learning. These parts are connected to a
- 20 loss layer 106 that receives the outputs of the said two parts 102, 104 and calculates the loss. In the image segmentation matching framework 100, three types of modules are used, namely a shared image feature extraction module 111, 112, 113, 114 for image feature extraction, for example realized as Resnet, a graph attention network (GAT) 120 for neighborhood graph feature description and a fully
- 25 connected neural network 116 for feature comparison.

- The shared image feature extraction module 110 is shown in Fig. 1 as separate modules 111, 112, 113, 114, corresponding to its functions in the basic model 102
- 30 and the graph-based regularization part 104. The input to the image segmentation matching framework 100 are image segmentations y_k 131 of a first image, image segmentations x_k 132 of a second image, and neighboring image segmentations 133 of the first image, which are specified by neighborhood graphs 134. The image

segmentations can be extracted from the full images using well known semantic or instances segmentations methods like DeepLab, Deep Parsing Network and Mask R-CNN. Here, it is assumed that the segmentation can be well obtained.

- 5 The segmentations of the vertices y_l of the neighborhood graphs G^{y_k} including information about the edges relating to the second image are input to the neighborhood graphs handling network 124. The information about the edges relating to the second image is handled by module 122, which outputs matrix $E(G^{x_k})$. $E(G^{x_k})$ is an all-ones matrix that is used to denote the adjacent matrices of G^{x_k}
- 10 representing the connection relationships between vertices in the corresponding neighborhood graphs. Spatial information may be obtained from sensors like radar, LiDAR, stereo camera and depth camera or even from other estimation methods like the monocular depth estimation networks. Here, there are no stringent requirements regarding the accuracy of the spatial information. The graph model is
- 15 constructed with edges connected between K-nearest neighbors in terms of spatial locations. In the scenarios of street scenes, a depth estimation method with 50 cm, 1 m, 2 m or even more meters accuracy may be sufficient.

- In the same way, segmentations y_k 131 of the first image, segmentations of the
- 20 vertices x_l 133 of the neighborhood graphs G^{x_k} including information about the edges relating to the second image are input to the third image feature extraction module 113 and the neighborhood handling network 124, respectively. For the sake of clarity, this path is not shown in Fig. 1, however it is taken into account in the descriptions and equations in the following. The first image and the second image
- 25 may contain similar contents. For example, they contain the same objects, captured from a different perspective and different ambient conditions such as daytime, seasonal conditions or light conditions. The typical application is a street scene in autonomous driving.

- 30 The first image feature extraction module 111 extracts the first image features $f(y_k)$ of the first segmentation y_k and outputs the extracted first image features $f(y_k)$ to the fully connected neural network 116. The second image feature extraction module 112 extracts the second image features $f(x_k)$ of the second segmentation x_k and

outputs the extracted second image features $f(x_k)$ also to the fully connected neural network 116. The fully connected neural network 116 compares the extracted first image features $f(y_k)$ with the extracted second image features $f(x_k)$ and outputs a predicted result $\hat{\alpha}_k$, for example an estimated classification, to the loss layer 106.

- 5 The loss layer 106 then calculates a total loss l_{total} as a sum of the loss l_{ce} based on the predicted result $\hat{\alpha}_k$ from the fully connected neural network 116 and a loss l_{reg} based on the result of the graph-based regularization part 104.

- 10 The graph-based regularization part 104 comprises a third image feature extraction module 113 and a neighborhood graphs handling network that extracts features of the neighboring segmentations or vertices $\{y_l\}$, which are elements of the neighborhood graphs G^{y_k} using the information about the edges of the graphs G^{y_k} . Descriptively said, neighborhood graphs handling network 124 including the graph attention (GAT) network 120 emphasizes the relevant neighboring segmentations of segmentation y_k , compares the resulting features with the extracted features of segmentation x_k in the discriminator layer of discriminator 108 and provides the result to the loss layer 106 for regulation. In particular, the graph-based regularization part 104 comprises a fourth image feature extraction module 114, a graph attention network 120, and a discriminator 108. The third image feature
- 15 extraction module 113 extracts third image features of the second segmentation x_k and outputs the extracted third image features $f(x_k)$ to the discriminator 108. The fourth image feature extraction module 114 extracts fourth image features $f(y_l)$ of neighboring segmentations contained in graph G^{y_k} of the first image segmentation y_k and outputs the extracted fourth image features $f(y_l)$ to the graph attention
- 20 network 120. The graph attention network 120 executes attention functions over the fourth image features and provides the output to the multi-head module 128, which are concatenated and averaged by the concatenation module 129 to obtain high dimensional features $\tilde{f}(y_c)$ of the neighboring segments. The high dimensional features $\tilde{f}(y_c) = g(G^{y_k})$ of the neighboring segmentations are finally output to the
- 25 discriminator 108 by the graph attention network 120. The discriminator 108 compares the extracted third image features $f(x_k)$ with the high dimensional features $g(G^{y_k})$ of the neighboring segmentations contained in graph G^{y_k} in layers of the discriminator 108 and outputs the result $L_{empirical-ID}(x, G^y)$ to the loss layer 106. The
- 30

loss layer subtracts $L_{\text{empirical-ID}}(\mathbf{x}, \mathbf{G}^y)$ from the loss l_{ce} obtained from the basic model 102 as to regulate the joint feature and metric learning of the basic model 102.

Fig. 3 shows a block diagram of an example of the structure of the image feature extraction module 110 in more detail. The module 110 is shared by the first, second, third, and fourth image feature extraction module 111, 112, 113, and 114, respectively, which therefore have the same structure. The reference sign 110 is therefore representative for any of the modules 111, 112, 113, and 114. The image feature extraction module 110 may be a Resnet, which comprises, after an input layer, a convolutional layer 301, a max pooling layer 302, several blocks 310 containing convolutional layers 301, and an average pooling layer 304. The structure may differ from the one shown. The Resnet 110 provides residual learning in which the subtraction of features is learnt from the input of that layer by using shortcut connections, that is, directly connecting input of (n)th layer to some (n+x)th layer, which is shown as curved arrow in Fig. 3. The residual learning improves the performance of model training, especially when the model is a deep network with more than 20 layers, and also resolves the problem of degrading accuracy in deep networks. The outputs from the Resnet are provided to the fully connected neural network 116, which may be a hidden-layers neural forward network.

Fig. 4 shows a block diagram of an example of the fully connected neural network 116 within the basic model 102 that receives the output of the Resnets 111 and 112. The fully connected neural network 116 may consist of several fully connected layers 401, each followed by a ReLU or LeakyReLU layer performing corresponding activation functions 402 and a final sigmoid layer 403 performing a sigmoid activation function. The fully connected neural network serves the decision-making. The input of the sigmoid function in the last layer is the squared difference between the two high-dimensional features of a pair of samples output from the Resnets 111 and 112. The output of the sigmoid function is the predicted result $\hat{\alpha}_k$, where k denotes the k-th pair of segmentation samples.

Fig. 5 shows a block diagram of an example of the discriminator 108. The discriminator 108 contains a discriminator layer 501 in form of a bilinear layer 501,

and a sigmoid layer 502. For the regularization, the high dimensional feature $f(x_k)$ of the vertex x_k is compared with the feature $g(G^{y_k})$ of the neighborhood graph G^{y_k} in the bilinear layer 501 with a trainable matrix $M \in \mathbb{R}^{n \times m}$:

$$d(a, b) = \sigma(a^T M b), \quad (3)$$

where $\sigma(\cdot)$ denotes the sigmoid function, a is a placeholder for $f(x_k)$ or $f(y_k)$ and b is a placeholder for $g(G^{y_k})$ or $g(G^{x_k})$. The following function discrimination function is defined:

$$L_{\text{pair-empirical-ID}} =$$

$$\begin{aligned} & \sum_k \frac{1}{k} \left\{ \mathbf{1}_{\{x_k \leftrightarrow y_k\}} \frac{1}{2} \left(\log[d(f(x_k), g(G^{y_k}))] + \log[d(f(y_k), g(G^{x_k}))] \right) \right. \\ & \quad \left. + \mathbf{1}_{\{x_k \nleftrightarrow y_k\}} \frac{1}{2} \left(\log[1 - d(f(x_k), g(G^{y_k}))] + \log[1 - d(f(y_k), g(G^{x_k}))] \right) \right\} \end{aligned} \quad (4)$$

$$\begin{aligned} & = \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{x_k \leftrightarrow y_k\}} \log[d(f(x_k), g(G^{y_k}))] + \mathbf{1}_{\{x_k \nleftrightarrow y_k\}} \log[1 - d(f(x_k), g(G^{y_k}))] \right\} \\ & + \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{y_k \leftrightarrow x_k\}} \log[d(f(y_k), g(G^{x_k}))] + \mathbf{1}_{\{y_k \nleftrightarrow x_k\}} \log[1 - d(f(y_k), g(G^{x_k}))] \right\} \end{aligned}$$

$$= L_{\text{empirical-ID}}(x, G^{y_k}) + L_{\text{empirical-ID}}(y, G^{x_k})$$

in which $\mathbf{1}_{\{\cdot\}}$ is the indicator, as well as $x_k \leftrightarrow y_k$ and $x_k \nleftrightarrow y_k$ denote the matching or mismatching relationship with respect to the k -th segmentation pair (x_k, y_k) , respectively. Since $L_{\text{empirical-ID}}(x, G^y)$ is similar to $L_{\text{empirical-ID}}(y, G^x)$ in a symmetrical way, in the theoretical analysis below, $L_{\text{empirical-ID}}(x, G^y)$ are simply denoted by $L_{\text{empirical-ID}}$.

Referring again to Fig. 1, the output $\hat{\alpha}_k$ of the fully connected neural network 116 and the output $L_{\text{pair-empirical-ID}}$ of the discriminator 108 are provided to the loss layer 106.

The loss function part computed by the loss layer for the fully connected neural network is the cross entropy given by

$$\ell_{ce} = -\frac{1}{k} \sum_k \{\alpha_k \log \hat{\alpha}_k + (1 - \alpha_k) \log(1 - \hat{\alpha}_k)\} \quad (1)$$

- 5 where k denotes the k -th pair of segmentation samples, α_k denotes the ground-truth label, that is “1” (one) for matched or “0” (zero) for unmatched label, and $\hat{\alpha}_k$ is the predicted result based on the sigmoid function ($\hat{\alpha}_k \in [0, 1]$).

10 Actually, $L_{\text{pair-empirical-ID}}$ is designed based on an information distance between the distributions conditioned by matched and unmatched pairs, which is explained in details in Proposition 1 below.

In the final loss function, the loss function part computed by the loss layer for the fully discriminator 108 is set to

$$\ell_{reg} = -L_{\text{pair-empirical-ID}}, \quad (5)$$

to maximize the information distance.

20 The final loss function is defined as

$$\ell_{total} = (1 - \lambda)\ell_{ce} + \lambda\ell_{reg}, \quad (2)$$

25 where $\lambda > 0$, is the predefined weight parameter. For example, $\lambda = 0.5$ may be chosen.

Fig. 2 shows a block diagram of an example of a Graph Attention Network (GAT) block 121. The GAT block 121 comprises a multi-head attention module, where the graph attention network is extended to multiple heads, which runs through an attention mechanism several times in parallel, to increase the expressiveness of the graph attention network, which is indicated in Fig. 1 by the indices 1...d...K. The

extracted high dimensional features $f(y_c)$ corresponding to $g(G^{y_k})$, which are in this case extracted from segmentations of neighbors of the segmentation y as defined by the output $E(G^{y_k})$ of module 122, are input to the attention function. The attention functions are softmax functions, where a non-linearity (here LeakyReLU) is applied before the softmax. E.g., for the first head, the head function can be expressed as $\text{head1} = \text{Attention}(W(r)f(y_c), W(r)f(y_d)) = \text{softmax}(a^T(W(r)f(y_c), W(r)f(y_d)))$ with a^T being a transposed weight vector a , $W(r)$ a weight matrix, and d runs from 1 to K . The result are attention coefficients η_{cd} that are multiplied with $W(r)f(y_d)$, concatenated and averaged.

Fig. 6 shows a flow diagram of a computer implemented method 600 for image segmentation matching of a first segmentation y_k of a first image with a second segmentation x_k of a second image, where the first segmentation y_k and second segmentation x_k are a pair of segmentations, comprising the steps:

learning 602 joint features and metric of the first segmentation y_k and the second segmentation x_k using joint feature and metric learning; and regulating 604 the joint feature and metric learning using a graph G^{y_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and neighboring segmentations of the first segmentation y_k ; and using the second segmentation x_k .

It is noted that the steps described herein may be performed in parallel or in another order where meaningful.

Fig. 7 shows a flow diagram based on the method 600 performed within the framework 100 and in particular within the basic model 102 presented in this disclosure. The flow diagram in Fig. 7 shows the part of the joint feature and metric learning 602 in more detail and uses the output of step 604 as input into the loss layer. The steps of the joint feature and metric learning 602 are the following: Extracting 702 first image features $f(y_k)$ of the first segmentation y_k by the first image feature extraction module 111 and outputting the extracted first image features $f(y_k)$ to the fully connected neural network 116.

Extracting 704 second image features $f(x_k)$ of the second segmentation x_k by the second image feature extraction module 112 and outputting the extracted second image features $f(y_k)$ to the fully connected neural network 116.

Comparing 706 the extracted first image features $f(y_k)$ and the extracted second image $f(x_k)$ features and outputting a predicted result $\hat{\alpha}_k$ to the loss layer 106, by the fully connected neural network 116.

A further step 708 is performed using the output of step 604:

Calculating 708 a total loss l_{total} as a sum of the loss l_{ce} based on the predicted result $\hat{\alpha}_k$ and a loss l_{reg} based on the result of the graph-based regularization part, by the loss layer 106.

Fig. 8 shows a flow diagram that based on the method 600 and that shows the step 604 in more detail. Also this part is performed within the framework 100, in particular within the graph-based regularization part 104. This method part comprises the following steps:

Extracting 802 third image features $f(x_k)$ of the second segmentation x_k by the third image feature extraction module 113 and outputting the extracted third image features $f(x_k)$ to the discriminator 108.

Extracting 804 fourth image features $f(y_l)$ of neighboring segmentations contained in graph G^{y_k} of the first image segmentation y_k by the fourth image feature extraction module 114 and outputting the extracted fourth image features $f(y_l)$ to the graph attention network 120.

Executing 806 attention functions over the fourth image features $f(y_l)$, in a multi-head module 128 to obtain high dimensional features $g(G^{y_k})$ of the neighboring segments, and outputting the high dimensional features $g(G^{y_k})$ of the neighboring segmentations to the discriminator 108, by the graph attention network 120, and Comparing 808 the extracted third image features $f(x_k)$ with the high dimensional features $g(G^{y_k})$ of the neighboring segmentations contained in graph G^{y_k} in layers of the discriminator 108 and outputting the result $L_{empirical-ID}(x, G^y)$ to the loss layer 106, by the discriminator 108.

In the following, a theoretical analysis is provided for Graph-Based Regularization.

Assumption 1. Let $(x, \mathcal{G}^y)$ be a random sample pair of an image segmentation and a neighborhood graph which are from the frames $\mathcal{F}^x$ and $\mathcal{F}^y$, respectively. As well, these random sample pairs are independent and identically distributed (i.i.d.). In this case, $(f(x), g(\mathcal{G}^y))$ denotes a pair of high-dimensional features output from Resnet and GAT respectively, which is corresponding to $(x, \mathcal{G}^y)$

With respect to $(f(x), g(\mathcal{G}^y))$, the random sample pairs are also i.i.d. Let $(\mathbb{A}, 2^{\mathbb{A}}, P_{f(x)g(\mathcal{G}^y)})$ be the probability space for $(f(x), g(\mathcal{G}^y))$, where $\mathbb{A}$ is the sample space of $(f(x), g(\mathcal{G}^y))$, $2^{\mathbb{A}}$ denotes the collection of all subsets of $\mathbb{A}$, as well as

$$P_{f(x)g(\mathcal{G}^y)}(\{(f(x), g(\mathcal{G}^y))\}) = p(f(x), g(\mathcal{G}^y)) \text{ with}$$

$$\sum_{(f(x), g(\mathcal{G}^y)) \in \mathbb{A}} p(f(x), g(\mathcal{G}^y)) = 1.$$

The matched conditional distribution and unmatched conditional distribution with respect to $f(x)$ and $g(\mathcal{G}^y)$ are denoted by $P_{f(x)g(\mathcal{G}^y)|x \leftrightarrow y}$ and $P_{f(x)g(\mathcal{G}^y)|x \nleftrightarrow y}$ whose densities are $p(f(x), g(\mathcal{G}^y)|x \leftrightarrow y)$ and $p(f(x), g(\mathcal{G}^y)|x \nleftrightarrow y)$ respectively, where the condition is whether the two image segmentations x and y are matched.

When the Assumption 1 is satisfied, we shall discuss some theoretical characteristics for the objective function optimization with respect to the graph-based regularization. We first state the expected form of $L_{\text{empirical-ID}}$ as

$$L_{\text{ID}} = \mathbb{E} [\mathbf{1}_{\{x \leftrightarrow y\}} \log[d(f(x), g(\mathcal{G}^y))]] + \mathbb{E} [\mathbf{1}_{\{x \nleftrightarrow y\}} \log[1 - d(f(x), g(\mathcal{G}^y))]]. \quad (6)$$

Proposition 1 (Relationship with KL divergence). In terms of the optimization for graph-based regularization, it is essential to update the parameters of neural networks by maximizing L_{ID} given by (6). Suppose that the optimal discriminator d^* is obtained and the corresponding objective function is denoted by $L_{\text{ID}}^{d^*}$. In this case,

maximizing L_{ID}^{d*} is equivalent to maximizing an upper bound of Kullback-Leibler (KL) divergence between $P_{f(x)g(\mathcal{G}^y)|x \leftrightarrow y}$ and $P_{f(x)g(\mathcal{G}^y)|x \leftrightarrow y}$.

Proposition 2 (Optimal solution for the regularization). As for the optimization problem with respect to the regularization, it can be described as

$$\max_{f,g,d} \{L_{ID}\},$$

where L_{ID} is given by Eq. (4), f and g denote the Resnet and GAT respectively, as well as d is the discriminator. In this case, the optimal solution is reached when the unmatched conditional distribution lies in the boundary of its range.

Besides, we also investigate how the optimization objective function of the regularization is influenced by the discriminator disturbance.

Proposition 3 (Effect of discriminator disturbance). Consider the case there exists a disturbance on discriminator d , that is $\tilde{d} = d + \varepsilon$, where ε is a small enough parameter. As for the optimization objective function L_{ID} given by Eq. (4), it is obtained $|L_{ID}^{\tilde{d}} - L_{ID}| = O(\varepsilon)$ where $L_{ID}^{\tilde{d}}$ denotes the objective function with $\tilde{d}$. Furthermore, when the optimal discriminator d^* is obtained as Eq. (8) whose corresponding object function is denoted by $L_{ID}^{d^*}$, we have $|L_{ID}^{\tilde{d}^*} - L_{ID}^{d^*}| = O(\varepsilon^2)$ where $L_{ID}^{\tilde{d}^*}$ denotes the objective function based on the optimal discriminator with a disturbance.

Moreover, we shall discuss how the mapping from vertices to neighborhood graphs makes an effect on the matching effectiveness in some special cases.

Case 1. There are two sampled frames $\mathcal{F}^1$ and $\mathcal{F}^2$ in which there exist the same vertices, implying that the image segmentations of objects and the corresponding neighborhood graphs are all common in the two frames. Let x and y denote the random vertices from the frames $\mathcal{F}^1$ and $\mathcal{F}^2$, as well as $\mathcal{G}^x$ and $\mathcal{G}^y$ are the neighborhood graphs with respect to x and y , respectively. Besides, $f(x)$ and $g(\mathcal{G}^x)$ are the high-dimensional features for x and $\mathcal{G}^x$, where $f(\cdot)$ and $g(\cdot)$ are bijective mappings represented by two neural networks.

Proposition 4 (Relationship with the mapping in Case 1)

When Case 1 is satisfied, if the mapping from vertices to their neighborhood graphs is injective, the correct image segmentation matching will be achieved for each pair of frames.

Case 2

As for a pair of frames $\mathcal{F}^1$ and $\mathcal{F}^2$, there exist some uncommon vertices so that their corresponding neighborhood graphs are also not the same, which is a different condition from that in Case 1. While, the rest conditions are similar to those in Case 1.

Proposition 5 (Relationship with the mapping in Case 2)

When Case 2 is satisfied, if there exists an injective mapping from vertices to their neighborhood graphs, we will have the correct results for the image segmentation matching.

Performance Evaluation

Performance of the KITTI Dataset. This dataset is available online, which provides a deal of multi-sensors data for autonomous driving. It contains street scene images and the corresponding LiDAR points. Generally speaking, the proposed framework 100, which is also referred to as REGR Net in the following, is superior to the other methods such as MatchNet, Siamese Network, TFeat Network, L2-Net, HardNet, SOSNet, or Res-Matching Network though there exist not too much difference between it and Res-Matching Network under the measurement of Recall. Moreover, it is also not difficult to observe that REGR Net performs less unstable than the other methods when the training state tends to convergence.

Actually, these methods are almost converged after around 15 epochs. As a result of the experiments, the proposed method performs better than other methods under the measurements of the whole criteria. As a result, the regularization proposed in this disclosure makes a positive difference to the matching efficiency by introducing graph-based information.

Performance of a real dataset

The real dataset is a dataset for autonomous driving applications, which is collected by a probe vehicle with many sensors including cameras, LiDAR, radars, etc.

- 5 Similar to KITTI dataset, it also contains images and LiDAR points, which are captured on the streets in Singapore as for the objects contained in an image, there are more numerous and more various landmarks such as traffic signs, traffic lights and poses in this dataset than those in KITTI dataset.
- 10 When taking experiments on the real dataset, the proposed method almost holds the optimal performance compared with the other methods, which is similar to that on KITTI dataset. Moreover, since the real dataset and KITTI dataset have different image quality and they are collected in different street scenes, there exists different performance in real data from that in KITTI data. That is, for all the discussed
- 15 methods, they performs better in real data. In general, the proposed method holds its superiority on the matching prediction in the case of high-quality image datasets and more amount of valuable objects.

- Other variations to the disclosed embodiments can be understood and effected by
- 20 those skilled in the art in practicing the claimed invention, from the study of the drawings, the disclosure, and the appended claims. In the claims, the word “comprising” does not exclude other elements or steps and the indefinite article “a” or “an” does not exclude a plurality. A single processor or other unit may fulfill the functions of several items or steps recited in the claims. The mere fact that certain
- 25 measures are recited in mutually different dependent claims does not indicate that a combination of these measures cannot be used to advantage. Any reference signs in the claims should not be construed as limiting the scope of the claims.

List of reference signs

	100	Framework
	102	Basic model
5	104	Graph-based regularization part
	106	Loss layer
	108	Discriminator
	110	Shared image feature extraction module, Resnet
	111	First image feature extraction module, Resnet
10	112	Second image feature extraction module, Resnet
	113	Third image feature extraction module, Resnet
	114	Fourth image feature extraction module, Resnet
	116	Fully connected layer
	120	GAT (graph attention) network module with GAT blocks
15	121	GAT block
	122	Module with Information of edges
	124	Neighborhood graphs handling network
	128	Multi-head module
	129	Concatenation module
20	131	Image segmentation of a first image
	132	Image segmentation of a second image
	133	Image segmentations of the first (or second) image
	134	Neighborhood graphs
	301	Convolution layer
25	302	Max pooling
	304	Average pooling
	310	Resnet convolution layer block
	401	Fully connected layer
	402	ReLU or LeakyReLU layer
30	403	Sigmoid layer
	501	Bilinear layer
	502	Sigmoid layer
	600	Method (flow diagram)

	602	First step of method part 1
	604	Second step of method part 1
	702	First sub-step of first step 602
	704	Second sub-step of first step 602
5	706	Third sub-step of first step 602
	708	Further step of method part 1
	802	First sub-step of second step 604
	804	Second sub-step of second step 604
	806	Third sub-step of second step 604
10		

Patent claims

1. Computer implemented method (600) for image segmentation matching of a first segmentation y_k of a first image with a second segmentation x_k of a second image, where the first segmentation y_k and second segmentation x_k are a pair of segmentations, comprising the steps:

learning (602) joint features and metric of the first segmentation y_k and the second segmentation x_k using joint feature and metric learning; and

regulating (604) the joint feature and metric learning using

a graph G^{y_k} containing a global spatial relationship between the first segmentation y_k corresponding to the second segmentation x_k and neighboring segmentations of the first segmentation y_k ; and using the second segmentation x_k .

and a graph G^{x_k} containing a global spatial relationship between the second segmentation x_k and neighboring segmentations of the second segmentation x_k .

2. Computer implemented method (600) according to claim 1, wherein the method is executed within a framework (100), the framework (100) comprising a basic model (102) for the joint feature and metric learning, a graph-based regularization part (104) for regulating the joint feature and metric learning, and a loss layer (106), wherein

the basic model (102) comprises a first image feature extraction module (111), a second image feature extraction module (112) and a fully connected neural network (116), wherein the method further comprises the steps:

Extracting (702) first image features $f(y_k)$ of the first segmentation y_k by the first image feature extraction module (111) and outputting the extracted first image features $f(y_k)$ to the fully connected neural network (116);

Extracting (704) second image features $f(x_k)$ of the second segmentation x_k by the second image feature extraction module (112) and outputting the extracted second image features $f(x_k)$ to the fully connected neural network (116);

Comparing (706) the extracted first image features $f(y_k)$ and the extracted second image $f(x_k)$ features and outputting a predicted result $\hat{\alpha}_k$ to the loss layer (106), by the fully connected neural network (116);

Calculating (708) a total loss l_{total} as a sum of the loss l_{ce} based on the predicted result $\hat{\alpha}_k$ and a loss l_{reg} based on the result of the graph-based regularization part, by the loss layer (106).

3. Computer implemented method (600) according to claim 2, wherein the graph-based regularization part (104) comprises a third image feature extraction module (113), a fourth image feature extraction module (114), a graph attention network (120), and a discriminator (108); wherein the method further comprises the steps:

Extracting (802) third image features $f(x_k)$ of the second segmentation x_k by the third image feature extraction module (113) and outputting the extracted third image features $f(x_k)$ to the discriminator (108);

Extracting fourth image features $f(y_l)$ of neighboring segmentations contained in graph G^{y_k} of the first image segmentation y_k by the fourth image feature extraction module (114) and outputting the extracted fourth image features $f(y_l)$ to the graph attention network (120);

Executing (804) attention functions over the fourth image features $f(y_l)$, in a multi-head module (128) to obtain high dimensional features $g(G^{y_k})$ of the neighboring segments, and outputting the high dimensional features $g(G^{y_k})$ of the neighboring segmentations to the discriminator (108), by the graph attention network (120); and

Comparing (806) the extracted third image features $f(x_k)$ with the high dimensional features $g(G^{y_k})$ of the neighboring segmentations contained in graph G^{y_k} in layers of the discriminator (108) and outputting the result $L_{empirical-ID}(x, G^y)$ to the loss layer (106), by the discriminator (108).

4. Computer implemented method (600) according to claim 2 or 3, wherein additionally the steps of claim 3 are performed with respect to image features $f(y_k)$ of the first segmentation y_k and image features $f(x_l)$ of neighboring segmentations contained in graph G^{x_k} of the second image segmentation x_k , and to compare third

image features $f(y_k)$ of the first segmentation (y_k) with the high dimensional features $g(Gx_k)$ of the neighboring segmentations in layers of the discriminator (108) and outputting the result $L_{\text{empirical-ID}}(y, G^x)$ to the loss layer (106), by the discriminator (108).

5

5. Computer implemented method (600) according to claim 3 or claim 4, wherein the discriminator (108) comprises a bilinear layer with a trainable matrix M.

6. Computer implemented method (600) according to claim 5, wherein the bilinear layer form is given as $d(a, b) = \sigma(a^T M b)$, where $\sigma(\cdot)$ denotes the sigmoid function, where, a^T is a weight vector, and a and b are placeholders for $f(x_k)$ or $f(y_k)$, wherein the discriminator computes a loss $L_{\text{pair-empirical-ID}}$ according to a discrimination function, which is defined as

$$L_{\text{pair-empirical-ID}} = L_{\text{empirical-ID}}(X, G^y) + L_{\text{empirical-ID}}(y, G^x) =$$

$$= \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{x_k \leftrightarrow y_k\}} \log[d(f(x_k), g(\mathcal{G}^{y_k}))] + \mathbf{1}_{\{x_k \nleftrightarrow y_k\}} \log[1 - d(f(x_k), g(\mathcal{G}^{y_k}))] \right\} \\ + \frac{1}{2} \frac{1}{k} \sum_k \left\{ \mathbf{1}_{\{y_k \leftrightarrow x_k\}} \log[d(f(y_k), g(\mathcal{G}^{x_k}))] + \mathbf{1}_{\{y_k \nleftrightarrow x_k\}} \log[1 - d(f(y_k), g(\mathcal{G}^{x_k}))] \right\},$$

wherein $\mathbf{1}\{-\}$ is an indicator and $x_k \leftrightarrow y_k$ or $x_k \nleftrightarrow y_k$ denotes a match or mismatch with respect to a kth segmentation pair (x_k, y_k).

7. Computer implemented method (600) according to claim 6, wherein the loss l_{reg} is calculated as loss $l_{\text{reg}} = -L_{\text{pair-empirical-ID}}$, by the loss layer (106).

25

8. Computer implemented method (600) according to any one of claims 2 - 7, wherein a loss function for calculating the loss l_{ce} is a cross entropie over pairs of segmentation samples using a ground-truth label and the predicted result from the basic model (102), and the result obtained from graph-based regularization part (104); wherein the total loss is the weighted sum of the losses l_{ce} and l_{reg} .

30

9. Computer implemented method (600) according to any one of the previous claims, wherein the loss function l_{ce} is defined by

$$l_{ce} = -\frac{1}{k} \sum_k \{ \alpha_k \log \hat{\alpha}_k + (1 - \alpha_k) \log(1 - \hat{\alpha}_k) \}$$

5 wherein k is a k -th pair of segmentation samples, α_k a ground-truth label, and $\hat{\alpha}_k$ is a predicted result from the basic model (102) based on a sigmoid function.

10. Computer implemented method (600) according to any of the previous claims, wherein the first image contains a street scene from a first perspective under
10 first environmental conditions and the second image contains a street scene from a second perspective under second environmental conditions.

11. Computer implemented method (600) according to any of the previous claims, wherein the first and the second image segmentations comprise spatial
15 information.

12. Computer implemented method (600) according to any of the previous claims, wherein the first (111), the second (112), the third (113), and the fourth (114) image feature extraction modules are one shared network (110), which serves as
20 feature descriptor function f and extracts the first, second, third and fourth image features as high dimensional features from the respective image segmentations.

13. Computer implemented method (600) according to the previous claim, wherein the shared network (110) is a residual neural network, ResNet.
25

14. Computer readable medium on which a computer program for image segmentation matching of a first segmentation of a first image with a second segmentation of a second image is stored, the computer program comprising a basic model module (102) configured to learn joint features and metric of a first
30 segmentation of a first image and a second segmentation of a second image, and a graph-based regularization part (104) configured to regulate the joint feature and metric learning using a graph G^{xk} containing a global spatial relationship between

the first segmentation and neighboring segmentations of the first segmentation and a graph G^{y^k} containing a global spatial relationship between the second segmentation and neighboring segmentations of the second segment.

- 5 15. Processing circuitry configured to run a computer program for image segmentation matching of a first segmentation of a first image with a second segmentation of a second image, the computer program comprising a basic model module configured to learn joint features and metric of a first segmentation, and a graph-based regularization part configured to regulate the joint feature and metric
- 10 learning using a graph G^{x^k} containing a global spatial relationship between the first segmentation and neighboring segmentations of the first segmentation and a graph G^{y^k} containing a global spatial relationship between the second segmentation and neighboring segmentations of the second segment.