Dual Task Learning by Leveraging Both Dense Correspondence and Mis-Correspondence for Robust Change Detection With Imperfect Matches

Jin-Man Park[*,†,1], Ue-Hwan Kim[*,2], Seon-Hoon Lee[3], Jong-Hwan Kim[3]
[1]KETI, [2]GIST, [3]KAIST
https://github.com/SAMMiCA/SimSaC

Abstract

Accurate change detection enables a wide range of tasks in visual surveillance, anomaly detection and mobile robotics. However, contemporary change detection approaches assume an ideal matching between the current and stored scenes, whereas only coarse matching is possible in real-world scenarios. Thus, contemporary approaches fail to show the reported performance in real-world settings. To overcome this limitation, we propose SimSaC. SimSaC concurrently conducts scene flow estimation and change detection and is able to detect changes with imperfect matches. To train SimSaC without additional manual labeling, we propose a training scheme with random geometric transformations and the cut-paste method. Moreover, we design an evaluation protocol which reflects performance in real-world settings. In designing the protocol, we collect a test benchmark dataset, which we claim as another contribution. Our comprehensive experiments verify that SimSaC displays robust performance even given imperfect matches and the performance margin compared to contemporary approaches is huge.

1. Introduction

Robust scene change detection (SCD) [52] plays a crucial role in various areas such as visual surveillance [33], anomaly detection [10], mobile robotics [43] and autonomous vehicles [30]. The SCD task aims to identify the changes in the current scene compared to the scene at different time steps. For storing past observations, the SCD task demands a database system. Numerous factors including view variation, illumination change, dynamic objects, different weather and camera jitter make the task challenging.

The SCD task in general requires visual place recognition (VPR) [39] to pair the current scene (query) with the scenes (reference) stored in a database system. After VPR, SCD analyzes the paired scenes for identification of changes. Contemporary approaches for SCD, however, assumes an ideal match of the scene between the current and the past time steps although observing the same scene with a perfect match hardly occurs in real-world applications. Thus, contemporary approaches would not display the reported performance when deployed to practical systems. Furthermore, the need for investigating the effect of imperfect matches for SCD and developing a robust neural architecture for pragmatic SCD lingers.

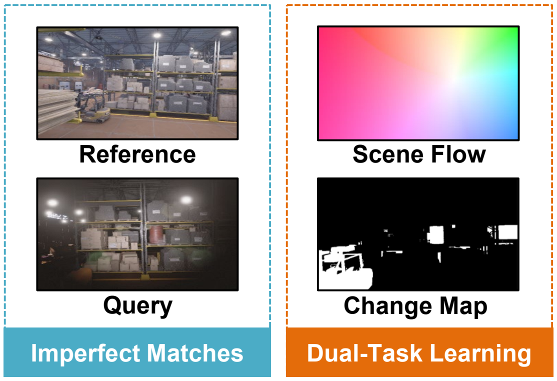

Figure 1. Overview of the proposed SimSaC. The proposed SimSaC displays robust performance even given imperfect matches of reference and query images with which conventional methods fail.

To overcome the limitation and develop a neural architecture that could readily function in real-world settings, we propose simultaneous Scene Flow Estimation and Change Detection (SimSaC). SimSaC is a neural network that concurrently performs scene flow estimation and change detection (Fig. 1). We propose to learn the two tasks in a multi-task learning setting (dual-task learning) by which SimSaC leverages dense correspondence (scene flow) and mis-correspondence (change). The proposed dual-task learning leads to robust change detection performance even given imperfect matches of reference and query images, in which

[*] These authors contributed equally to this work.
[†] work done while with KAIST

conventional methods are unsuccessful. In addition, we propose a training scheme to learn the two tasks without requiring additional labels. We introduce random geometric transformations [41] and the cut-paste method [19] to query images and generate pseudo labels for scene flow estimation and change detection.

Next, we carefully design an evaluation protocol for scene change detection that reflects performance in real-world settings and compare conventional methods under the proposed evaluation protocol. Specifically, we propose to measure performance with imperfect matches of reference and query images in addition to the conventional evaluation protocol. For this, we collect a new evaluation benchmark dataset. To collect the benchmark dataset, we first run an off-the-shelf VPR algorithm over change detection evaluation datasets and collect imperfect matches. Moreover, we attentively remove the incorrect matches where the VPR algorithm fails to discover the matches. We open-source the proposed benchmark dataset.

In summary, the main contributions of our work are as follows:

1. **Problem Formulation**: We carefully formulate a change detection task that reflects performance in real-world settings for the first time. We expect our work would provoke the evolution of the change detection task towards practical use cases.

2. **SimSaC Architecture**: We propose the SimSaC network for robust change detection which leverages both dense correspondence (scene flow) and mis-correspondence (change).

3. **Robust Training Scheme**: We design a training scheme that enhances the robustness of change detection without requiring additional annotations.

4. **Evaluation Dataset**: We collect a new benchmark dataset consisting of imperfect matches for measuring change detection performance in real-world scenarios.

5. **Open Source**: We contribute to the research society by making the source code of the proposed SimSaC network, the pretrained network parameters and the benchmark dataset public.

2. Related Works

2.1. Change Detection

Traditional Approaches. Traditional change detection approaches fall into two categories [23]: pixel-based and object-based approaches. First, pixel-based approaches handle a pair of images and extract features for detecting changes assuming perfect image registration between the paired images. Pixel-based approaches are inherently sensitive to noise and mis-registration errors; researchers have attempted to overcome the sensitiveness through diverse transformations. Principle component analysis (PCA) [16], iterative reweighted multivariate alteration detection (IR-MAD) [46] and Wavelet transformation [9] exemplify such attempts. Moreover, object-based approaches focus on meaningful objects rather than all pixels for change detection. Thus, object-based approaches in general demand segmentation algorithms. Multiple research efforts have enhanced the performance of object-based approaches. For instance, neighborhood correlation image analysis [28], parcel-based context-sensitive change detection [7] and object-level progressive change feature classification [25] have led the enhancement. However, traditional approaches have naturally faded away due to the dramatic performance enhancement of deep learning-based approaches.

Deep Learning Approaches. Deep learning-based approaches have become feasible after the introduction of several large-scale datasets. For example, Sakurada and Okatani have built TSUNAMI and Google Street View datasets [56] and Alcantarilla *et al.* VL-CMU-CD dataset [2]. Based on these datasets, various deep neural architectures have emerged: Nguyen *et al.* [45] have developed a triplet CNN architecture that extracts pertinent features for change detection; Chen *et al.* [12] have introduced an attention ConvLSTM architecture for performing pixel-level change detection; Mandal *et al.* [40] have designed spatio-temporal feature learning framework using 3D-CNN; Akilan *et al.* [1] have proposed a 3D-CNN LSTM architecture for detection pixel-wise changes over time; and Bakkay *et al.* [5] have utilized generative adversarial learning frameworks to extract motion features for performing change detection. These deep learning-based approaches, however, assume a perfect match between the query and reference images. Thus, these approaches would not guarantee the reported performance with imperfect matches which occur much more frequently than the assumed perfect matches in real-world settings.

2.2. Image Warping

Optical Flow. The optical flow estimation task estimates 2D scene flow fields between paired images. Dosovitskiy *et al.* [18] have proposed the first trainable CNN architecture, FlowNet which resembles a U-Net architecture. Next, various enhanced architectures such as FlowNet2 [27], SpyNet [53], PWC-NET [60] and LiteFlowNet [24] have followed. These architectures have either combined conventional architectures or employed feature pyramid networks for performance improvement. Nonetheless, these architectures display poor performance when strong geometric transformations appear or when the visual appearance significantly differs.

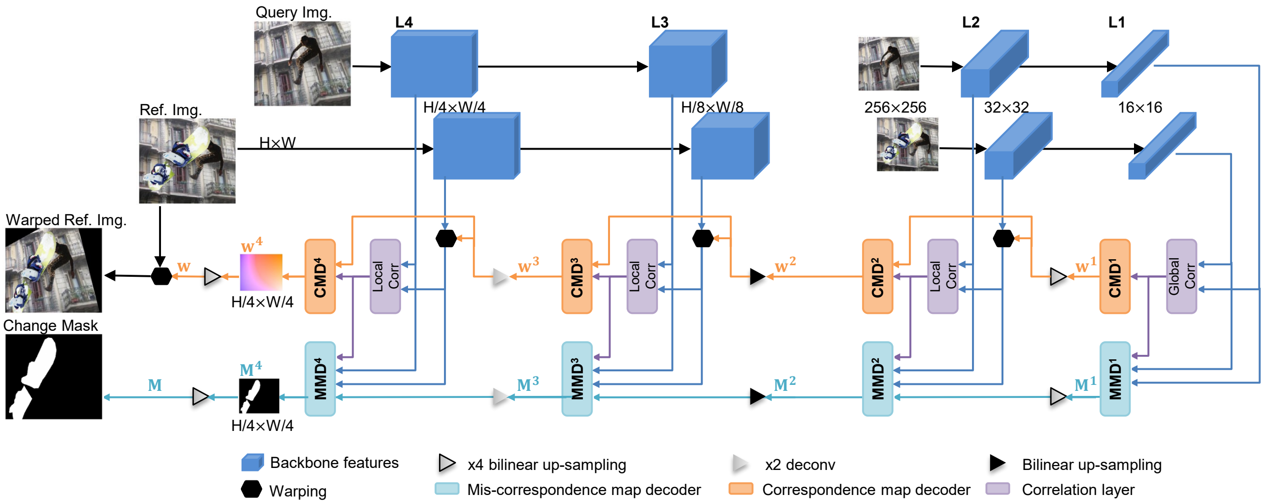

Figure 2. Architecture of the proposed SimSaC architecture. SimSaC consists of four major components: 1) a two-stream feature pyramid, 2) correlation layers estimating pixel-level similarity scores given a feature pair, 3) correspondence map decoder (CMD) estimating the scene flow w, and 4) mis-correspondence map decoder (MMD) estimating the change mask M.

Geometric Correspondence. In this work, we focus on dense correspondence within geometric correspondence. Geometric correspondence aims to robustly discover correspondence between two images even when large geometric displacement occurs. Melekhov *et al.* [41] have introduced DGC-Net inspired by the advances in optical flow estimation architectures for dense 2D correspondence. Next, Rocco *et al.* [54] have designed NC-Net that filters out ambiguous matches and retains matches satisfying cyclic consistency. Moreover, Truong *et al.* [61] have proposed GLU-Net for handling large and small displacements and dealing with any input resolution.

2.3. Visual Place Recognition (VPR)

VPR requires powerful image representations for image matching. In general, VPR utilizes two types of representations: global image descriptors and local keypoint descriptors. First of all, global image descriptors represent an image by one single feature vector. Early global image descriptors aggregated local keypoint descriptors through Bag of Words (BoW) [14, 59], Fisher Vectors (FV) [29, 50] or Vector of Locally Aggregated Descriptors (VLAD) [4, 31] for image representations. Recently, these approaches have employed deep neural networks for boosting performance. Examples of such approaches include NetVLAD [3], Patch-NetVLAD [22], NetBoW [48] and NetFV [42].

On the other hand, local keypoint descriptors describe salient regions of an image. Traditionally, hand-crafted features such as SIFT [38], SURF [6] and ORB [55] were prevalent. After the surge of deep learning, data-driven local features including LIFT [62], DeLF [47], SuperPoint [17] have appeared.

3. Methodology

3.1. Network Architecture

Overview. Fig. 2 displays the proposed SimSaC network architecture. SimSaC receives imperfect matches of query and reference images. Then, SimSaC performs dual tasks using two decoders: correspondence (scene flow) map decoder (CMD) and mis-correspondence (change) map decoder (MMD). Utilizing both dense correspondence and mis-correspondence allows warping of the reference image in respect of the query image. This dual-learning task in addition to the proposed training scheme detailed in the following subsection helps SimSaC achieve robust performance even given imperfect matches.

Two-Stream Feature Pyramid. SimSaC is a two-stream feature pyramid architecture [61] consisting of four levels (L=4)—enabling dense correspondence and mis-correspondence estimation for any input resolution. $L1$ and $L2$ deal with the query and reference images down-scaled to a fixed resolution $H_L \times W_L$ while $L3$ and $L4$ directly handle the original image resolution $H \times W$. $L3$ up-scales the features $L2$ has processed and $L4$ recovers the original resolution.

Correlation Layers. The local correlation layers [27] evaluate the feature correlation c^l between the query $F_q^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ and the reference $F_r^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ feature maps within a search radius R as follows:

$$c^l(\mathbf{x}, \mathbf{d}) = F_q^l(\mathbf{x})^T F_r^l(\mathbf{x} + \mathbf{d}), \quad ||\mathbf{d}||_\infty \leq R, \qquad (1)$$

where $\mathbf{x} \in \mathbb{Z}^2$ represents a feature map coordinate, $\mathbf{d} \in \mathbb{Z}^2$ the displacement from $\mathbf{x}$, and l the level in the feature pyramid. The dimensionality of c^l is $H_l \times W_l \times (2R+1)^2$. Furthermore, the global correlation layer [27] evaluates the feature correlation $C^l \in \mathbb{R}^{H_l \times W_l \times (H_l W_l)}$ between the query and the reference feature maps in all locations as follows:

$$C^l(\mathbf{x}, \mathbf{x}') = F_q^l(\mathbf{x})^T F_r^l(\mathbf{x}'). \tag{2}$$

Remark. We note that the components of the proposed SimSaC architecture appeared in previous works; however, any single design choice cannot explain the superiority of the proposed SimSaC architecture, but their composition. Thus, one's presumably considering our results as findings rather than a novel algorithm does not diminish the importance of our work.

3.2. Correspondence Map Decoder (CMD)

Correspondence map decoder (CMD) aims to find pixel-wise dense correspondence between a query $I_q \in \mathbb{R}^{H \times W \times 3}$ and a reference $I_r \in \mathbb{R}^{H \times W \times 3}$. Dense correspondence often referred to as scene flow $\mathbf{w} \in \mathbb{R}^{H \times W \times 2}$ warps I_r towards I_q as follows:

$$I_q(\mathbf{x}) \approx I_r(\mathbf{x} + \mathbf{w}(\mathbf{x})). \tag{3}$$

The flow $\mathbf{w}$ depicts the pixel-wise 2D motion in the coordinate system of the query image and has a direct connection with the pixel correspondence map $\mathbf{m}(\mathbf{x}) = \mathbf{x} + \mathbf{w}(\mathbf{x})$.

Given an image pair and the ground truth pixel correspondence map w_{gt}, we use a hierarchical end-point error (EPE) for training CMD as follows:

$$\mathcal{L}_c = \sum_{l=0}^{L-1} \alpha^{(l)} \frac{1}{N^{(l)}} \sum_{i=0}^{N^{(l)}-1} \|\hat{w}_i^{(l)} - w_i^{(l)}\|_1, \tag{4}$$

where $\|.\|_1$ is the L1 distance between an estimated dense correspondence map $\hat{w}^{(l)}$ and the ground truth one $w^{(l)}$; i indexes over pixel locations $N^{(l)}$ at each level l of the L-level feature pyramid. In order to adjust the weight of different pyramid layers, we introduce a vector of scalar weight coefficients $\alpha^{(l)}$.

3.3. Mis-Correspondence Map Decoder (MMD)

Mis-correspondence map decoder (MMD) targets to estimate the pixel-level mis-correspondence between a query $I_q \in \mathbb{R}^{H \times W \times 3}$ and a reference $I_r \in \mathbb{R}^{H \times W \times 3}$. We represent the mis-correspondence map as a probability map whose pixel values indicate the mis-correspondence score at each pixel. Given an image pair and the ground-truth pixel mis-correspondence score map, we optimize a hierarchical focal loss for dealing with an imbalanced pixel classification problem as follows:

$$\begin{aligned} e_i^{(l)} &= (1 - \hat{M_i}^{(l)})^{\gamma} M_i^{(l)} log(\hat{M_i}^{(l)}) \\ &\quad + (\hat{M_i}^{(l)})^{\gamma} (1 - M_i^{(l)}) log(1 - \hat{M_i}^{(l)}), \\ \mathcal{L}_m &= - \sum_{l=0}^{L-1} \beta^{(l)} \frac{1}{N^{(l)}} \sum_{i=0}^{N^{(l)}-1} e_i^{(l)}, \end{aligned} \tag{5}$$

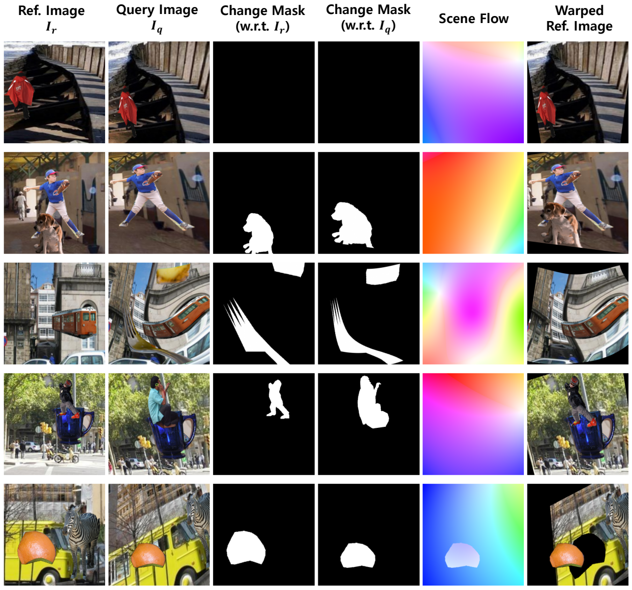

Figure 3. Generated synthetic samples for dual-task learning of both dense correspondence and mis-correspondence.

where $e_i^{(l)}$ is the focal loss for a pixel indexed as i, $M_i^{(l)}$ and $\hat{M_i}^{(l)}$ are the ground-truth and estimated mis-correspondence scores of a pixel, respectively, and γ is the scalar constant for reducing the relative loss for well-classified pixels ($\gamma = 0.5$). We also use a vector of scalar weight coefficients $\beta^{(l)}$.

3.4. Loss Function

The objective function $\mathcal{L}$ for SimSaC combines the correspondence loss $\mathcal{L}_c$ and the mis-correspondence loss $\mathcal{L}_m$ as follows:

$$\mathcal{L} = \mathcal{L}_c + \lambda \mathcal{L}_m, \tag{6}$$

where λ is a weight coefficient ($\lambda = 1.0$). Details on the hyper-parameters used for training and precise network definitions of all network components follow in the supplementary material.

3.5. Training Scheme for Dual-Task Learning

Training the proposed SimSaC network requires labels of scene flow and change mask for each image pair. However, there is no available dataset providing both types of labels and annotating such labels costs a lot of time and manual work. To cope with the limitation, we propose to generate training samples for dual-task learning of dense correspondence and mis-correspondence using random image warping [41] and the cut-paste method [19] (Fig. 3). To guarantee the diversity of training samples and the generality of the models, we devise five types of samples, each corresponding to the rows of Fig. 3, as follows:

- **Static** (row 1): Static pairs do not have miscorrespondence between query and reference images though a correspondence map exists. We attach random foreground objects to a reference image, construct a random scene flow map, and warp the reference image with the attached foreground objects using the scene flow map to create a pair of reference and query images.

- **Missing** (row 2): In this class of image pairs, some foreground objects appear on reference images but not on the corresponding query images. The rest of the sample generation process is identical to that of "static".

- **New** (row 3): In this class of image pairs, some foreground objects appear on query images but not on the corresponding reference images.

- **Replaced** (row 4): In this class of image pairs, the foreground objects on reference images get replaced on the corresponding query images. We choose foreground objects that have similar areas as the original objects for replacement.

- **Moved** (row 5): In this class of image pairs, the foreground objects on reference images get translated and rotated on the corresponding query images.

4. Experiments

4.1. Datasets and Preprocessing

We employ four datasets for training and validation of the proposed SimSaC architecture: Synthetic, ChangeSim [49], VL-CMU-CD [2] and PCD [56]. We use the Synthetic dataset for pretraining or training processes while we adopt other three datasets for both training and testing. Further, we apply the proposed training scheme for dual-task learning only to Synthetic for fair comparison.

Synthetic amounts to 200,000 image pairs generated following the proposed training scheme for dual-task learning. The number of samples for each class ("static", "missing", "new", "replaced" and "moved") is 40,000. For reference images, we retrieve images from the DPED [26], Cityscapes [13], and ADE-20K [63] datasets and apply the proposed training scheme to construct query images. Moreover, we utilize the COCO [36] for foreground objects.

ChangeSim is a photo-realistic indoor dataset created using Unreal Engine 4 [32]. It consists of 20 image sequences (approximately 1,000 frames for each sequence) taken in ten different environments (two sequences from each environment). Among the 20 sequences, 12 sequences are for training and 8 sequences for testing. Each query image is roughly matched with the corresponding reference image using its estimated 7-D pose obtained by RTAB-MAP [34], one of the latest visual SLAM algorithms. ChangeSim provides three versions of visual variation for each image sequence: normal, dusty-air, and low-illumination. *Normal* is a situation where there is no visual change except for object changes in a pair, i.e., the air turbidity or light intensity between the two images remains the same. In *dusty-air* and *low-illumination*, query images become unclear compared to the reference images, i.e., the air turbidity increases in *dusty-air*, and the light intensity decreases in *low-illumination*. Each pair has a change mask with five change classes, i.e., static, new, missing, rotated and replaced. Since the multi-class change detection is beyond the scope of our approach, we ignore the change classes and treat the change masks as binary ones (static or changed). Note that we use only the normal split for training following the convention [49].

VL-CMU-CD is a long time span street-view scene change detection dataset containing 151 image sequences (approximately 9 frames for each sequence). Following the official splits [2], a total of 1,362 image pairs is split into 933 training pairs (98 sequences) and 429 test pairs (54 sequences). Each pair has a change mask with five semantic classes. Since the semantics are beyond the scope of our approach, we ignore the semantic classes and treat the change masks as binary ones (static or changed).

PCD consists of 200 pairs of panoramic images with the resolution of 224×1024 and hand-labeled change masks. The PCD dataset is divided into two subsets: GSV and TSUNAMI, where their domains are street views and post-tsunami scenes, respectively. Following the convention, we collect 224×224-sized patches by sliding 56 pixels in the horizontal direction and applying the data augmentation of plane rotation, resulting in a total of 24,000 image pairs. We adopt 5-fold cross-validation for model training and testing following the convention [57].

Dataset		Recall@1	Recall@5	Recall@10	Recall@20
VL-CMU-CD		76.8	87.7	89.8	91.7
PCD	GSV	85.8	87.5	87.8	88.1
	TSUNAMI	53.7	55.5	55.9	56.2
ChangeSim	Normal	66.7	68.9	69.6	72.0
	Dusty-air	47.8	50.4	51.6	53.4
	Low-illumi.	49.2	55.6	57.1	59.4

Table 1. VPR performance (Recall@K) of Patch-NetVLAD on the scene change detection datasets.

4.2. Preparation of Imperfect Matches

In order to propose and evaluation benchmark and show the robustness of the proposed model given an imperfect match, we generate imperfect matches from ChangeSim, VL-CMU-CD and PCD using the latest VPR model, Patch-NetVLAD [22]. The weights learnt from the Mapillary dataset [44] are used and fixed for all experiments. Recall@1 and Recall@5 of PatchNetVLAD for the Mapillary dataset are 79.5% and 86.2%, respectively.

Annotation of valid matching. If VPR retrieves a completely wrong image, the subsequent scene change detection is meaningless. To prevent this, we label the retrieved images by checking whether the extraction is valid. For ChangeSim, we consider retrieved images with a distance of less than 1 m from their query images as correct. For VL-CMU-CD, if the retrieved image and the query image belong to the same video, the retrieved image is considered to be correct. For PCD, the retrieved image is considered correct if the retrieved image and query image belong to the same panoramic image and their intersection over union (IoU) is greater than 0.6. Table 1 shows the quality of matches generated using VPR for SCD datasets evaluated through the Recall@K metric.

4.3. Implementation Details

Training Schedule. We employ two types of training schedules: target data only training (*T*) and 2-stage training ($S \rightarrow (S, T)$) that we newly propose as follows:

- *T*: Target data only training utilizes just one target dataset (either ChangeSim, VL-CMU-CD or PCD) for training.
- $S \rightarrow (S, T)$: The first stage of 2-stage training learns the dual-task from the Synthetic dataset for 25 epochs; and the second stage utilizes both the Synthetic and one target dataset for 25 epochs. To deal with the data imbalance problem, we down-sample the number of the Synthetic dataset to the number of the target dataset in the second stage. Moreover, we set $\mathcal{L}_c = 0$ for target datasets in the second stage since the target datasets do not provide ground-truth scene flows.

Data Augmentation. In the training phase, we randomly select one image in a pair and apply random image transforms such as color jittering, channel shuffling, grayscaling, Gaussian blurring, motion blurring, and random shadowing. We implement the data augmentations with the Albumentation library [8].

Evaluation Metrics. We evaluate performance using two metrics: the standard F1-score and the (average) performance drop. The F1 score is the harmonic mean of precision and recall as follows:

$$F1\text{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall}, \tag{7}$$

where $Precision = TP/(TP + FP)$, and $Recall = TP/(TP + FN)$; *TP*, *FP*, and *FN* are the number of true positives, false positives and false negatives, respectively, which are counted for every pixel. Next, we measure performance drop in percent between the change detection performances given ground-truth matching and imperfect matching.

Parameter Setting and Hardware. We initialized models from the ImageNet-pretrained weights and the weights of newly added layers by the Xavier method [20]. In the training phase, we optimized models using the AdamW optimizer [37] with the batch size of 16, and learning rate decay of 4×10^{-4}. The initial learning rate of 10^{-4} is halved at epochs 12, 20, and 23, respectively. All images are resized to 520×520 during training. We implemented our model with Pytorch. We trained and tested the models on a workstation with one NVIDIA RTX 3090 GPU with 24GB memory and one Intel i9 CPU.

Model	Backbone	Training	F1-score (%)						avg. PD (%)
			Normal		Dusty-air		Low-illumi.		
			GT	VPR	GT	VPR	GT	VPR	
CSCDNet [57]	Res18	*T*	32.6	14.7	15.5	11.5	13.5	2.6	53.4
CSCDNet [57]	Res18	*S→(S,T)*	30.1	17.6	14.2	10.4	13.2	4.2	44.2
DR-TANet [11]	Res18	*T*	40.2	25.7	22.0	16.4	17.7	6.4	39.4
DR-TANet [11]	Res18	*S→(S,T)*	38.1	24.5	20.4	15.6	17.1	4.7	40.9
Ours	VGG16	*T*	**69.1**	27.4	29.3	22.1	27.4	20.4	44.5
Ours	VGG16	*S→(S,T)*	66.5	**54.1**	**56.8**	**42.5**	**42.7**	**32.4**	**22.3**

Table 2. Quantitative results on the ChangeSim dataset. The best two results are marked in bold and underline, respectively.

Model	Backbone	Training	F1-score (%)		PD (%)
			GT	VPR	
EFNet [15]	U-Net	*T*	58.1	-	-
Siam-Cone [15]	U-Net	*T*	66.4	-	-
Siam-Diff [15]	U-Net	*T*	62.5	-	-
CosimNet [21]	DeepLabV2	*T*	70.6	-	-
CDNet [58]	U-Net	*T*	68.5	-	-
HPCFNet [35]	VGG-16	*T*	75.2	-	-
CSCDNet [57]	Res18	*T*	71.0	61.0	14.1
CSCDNet [57]	Res18	*S→(S,T)*	69.0	63.0	8.7
DR-TANet [11]	Res18	*T*	75.5	62.6	17.1
DR-TANet [11]	Res18	*S→(S,T)*	72.1	60.9	15.5
Ours	VGG16	*T*	75.6	68.2	9.8
Ours	VGG16	*S→(S,T)*	**79.7**	**75.4**	**5.4**

Table 3. Quantitative results on the VL-CMU-CD Dataset. The best two results are marked in bold and underline, respectively.

Model	Backbone	Training	F1-score (%)						avg. PD (%)
			GSV		TSUNAMI		Average		
			GT	VPR	GT	VPR	GT	VPR	
EFnet [15]	U-Net	*T*	56.5	-	65.9	-	61.2	-	-
Siam-Cone [15]	U-Net	*T*	63.8	-	70.9	-	67.4	-	-
Siam-Diff [15]	U-Net	*T*	64.7	-	71.7	-	68.2	-	-
CosimNet [21]	DeepLabV2	*T*	69.2	-	80.6	-	74.9	-	-
CDNet [58]	U-Net	*T*	69.3	-	83.8	-	76.6	-	-
CDNet++ [51]	VGG19	*T*	68.0	-	86.0	-	77.0	-	-
HPCFNet [35]	VGG16	*T*	77.6	-	86.8	-	82.2	-	-
CSCDNet [57]	Res18	*T*	73.8	58.6	85.9	74.3	79.9	66.5	16.8
CSCDNet [57]	Res18	*S→(S,T)*	69.1	57.8	75.7	65.9	72.4	61.9	14.6
DR-TANet [11]	Res18	*T*	72.9	56.1	88.6	75.5	80.8	65.8	18.5
DR-TANet [11]	Res18	*S→(S,T)*	68.6	55.1	74.1	65.2	71.4	60.2	15.7
Ours	VGG16	*T*	78.2	61.0	86.5	69.7	82.3	65.4	20.6
Ours	VGG16	*S→(S,T)*	**78.4**	**69.3**	**90.4**	**84.4**	**84.4**	**76.8**	**9.0**

Table 4. Quantitative results on the PCD Dataset. The best two results are marked in bold and underline, respectively.

4.4. Comparative Studies

We compare our model SimSaC with nine baselines: EFNet [15], Siam-Cone [15], Siam-Diff [15], CosimNet [21], CDNet [58], CDNet++ [51], HPCFNet [35], CSCDNet [57] and DR-TANet [11]. We evaluate each model with two types of matches: **GT** where we use ground-truth matches, and **VPR** where we use imperfect matches obtained by VPR. Note that we use the VPR pairs only for evaluation and the GT pairs for training. In addition, we trained CSCDNet and DR-TANet, which are open-sourced algorithms, using the two training schedules delineated in 4.3 and report their performance in the cases of **GT** and **VPR**. For other baselines, we report their reported performance evaluated on the ground-truth pairs due to the unavailability of their sources.

Evaluation on ChangeSim. Table 2 displays the quantitative comparison results on the ChangeSim dataset. The results attest that SimSaC outperforms the baselines with a large margin given GT pairs in every split (normal, dusty-air, and low-illumi.). Furthermore, the performance gap even increased as we fed imperfect pairs from VPR, while the baselines show a significant performance degradation given VPR pairs. The performance drop of SimSaC is only half of the baselines. Another major difference between SimSaC and other comparative models is the effect of the synthetic dataset. While the baselines show little or no performance improvement due to pre-training using the Synthetic dataset, SimSaC reveals a dramatic performance improvement. This implies learning the scene flow is the key to improving the change detection performance since SimSaC alone learns the scene flow estimation when learning with the Synthetic dataset.

Evaluation on VL-CMU-CD and PCD. Tables 3 and 4 present the quantitative comparison results on the VL-CMU-CD and PCD datasets, respectively. The results are consistent with those from ChangeSim. Specifically, SimSaC with the 2-stage training schedule introduced in our work exhibits exceedingly slight performance drop compared to the baselines. Further, the performance gap is larger in the case of ChangeSim compared to the VL-CMU-CD and PCD cases since VPR performance is lower for ChangeSim (Table 1)—establishing the effectiveness of SimSaC.

Qualitative Comparison. Fig. 4 depicts qualitative comparison results. From top to bottom, each row represents results from ChangeSim-normal, ChangeSim-dusty-air, ChangeSim-low-illumination, VL-CMU-CD, PCD-GSV, and PCD-TSUNAMI, respectively. The qualitative results corroborate that SimSaC can capture change regions robustly under the challenging conditions of large viewpoint variations and imperfect matches.

Training	F1-score (%)					
	ChangeSim			VL-CMU-CD	PCD	
	Normal	Dusty-air	Low-illumi.		GSV	TSUNAMI
T	**69.2**	29.4	27.5	75.6	$\underline{78.2}$	86.5
S	57.8	29.4	26.4	44.3	31.2	75.1
S→*T*	$\underline{67.3}$	42.8	32.3	$\underline{79.1}$	78.0	$\underline{89.9}$
(S,T)	64.1	$\underline{51.3}$	$\underline{37.2}$	67.5	77.1	87.0
S→*(S,T)*	66.5	**56.9**	**42.7**	**79.7**	**78.4**	**90.4**

Table 5. Ablation study of the training scheme. The best two results are marked in bold and underline, respectively.

Task	Training	F1-score (%)					
		ChangeSim			VL-CMU-CD	PCD	
		Normal	Dusty-air	Low-illumi.		GSV	TSUNAMI
C	*T*	**69.2**	29.4	27.5	75.6	78.2	86.5
C	*S*→*(S,T)*	$\underline{68.2}$	$\underline{45.1}$	$\underline{32.8}$	$\underline{77.1}$	**78.5**	$\underline{90.3}$
A→*C*	*S*→*(S,T)*	64.5	40.9	29.1	71.5	72.5	82.6
(A,C)	*S*→*(S,T)*	66.5	**56.9**	**42.7**	**79.7**	$\underline{78.4}$	**90.4**

Table 6. Ablation study of the dual-task learning. The best two results are marked in bold and underline, respectively.

4.5. Ablation studies

Impact of Training Schedule. Table 5 demonstrates the effectiveness of the proposed training schedule by comparing five different training schedules: T (target data only training), S (Synthetic only training), $S \rightarrow T$ (pretraining on Synthetic and finetuning on target data), (S, T) (joint training on Synthetic and target data), and $S \rightarrow (S, T)$ (2-stage training). Except for the normal split of the ChangeSim dataset, the proposed 2-stage training—which jointly trains on the target and Synthetic datasets after pretraining on the Synthetic dataset—shows the best performance. The result suggests that catastrophic forgetting for scene flow estimation occurs when fine-tuning only on the target dataset where there is no scene flow supervision. In other words, we could prevent catastrophic forgetting and retain the benefit of pretraining by jointly training on the target and the Synthetic datasets in the second stage.

Impact of Dual-Task Learning. Table 6 verifies the advantage of the proposed dual-task learning by comparing four different learning settings: C (change detection only) with the T schedule, C with the $S \rightarrow (S, T)$ schedule, $A \rightarrow C$ (image alignment, i.e., scene flow estimation, followed by change detection) with the $S \rightarrow (S, T)$ schedule, and (A, C) (dual-task learning) with the $S \rightarrow (S, T)$ schedule. The results indicate that performing scene flow estimation and change detection simultaneously exhibits consistently better performance in all cases than performing change detection alone. Note that, if scene flow estimation and change detection are performed sequentially ($A \rightarrow C$) instead of at the same time, it does not help to improve performance. This means that dual-task learning is the key to improving change detection performance.

5. Discussion and Conclusion

Limitation. Particularly, SimSaC could learn to distinguish irrelevant pairs rather than assuming pairs with at least

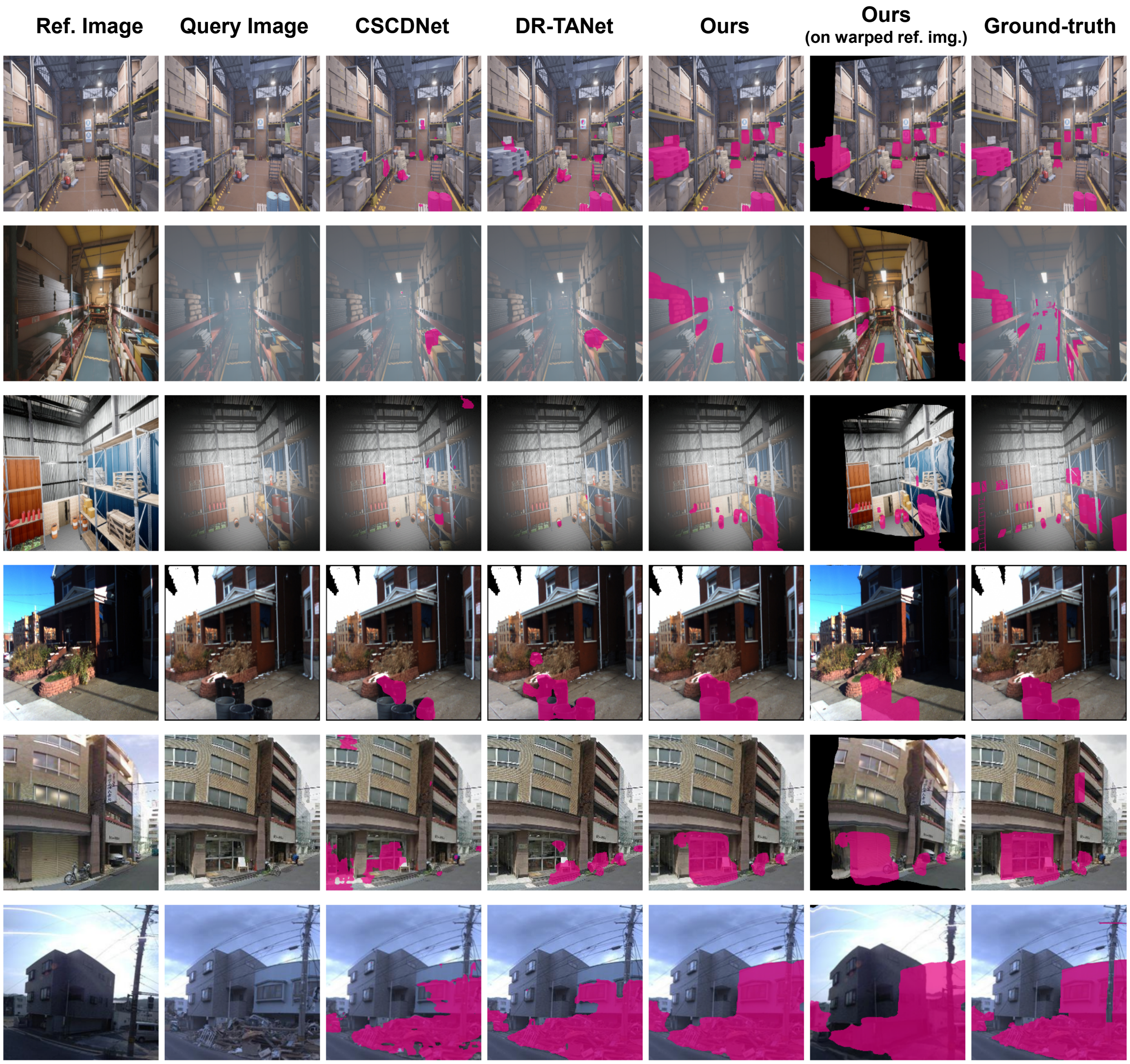

Figure 4. Qualitative results. The change masks are marked with purple. Our SimSaC effectively handles large variation in camera viewpoints and imperfect matches. Refer to the supplementary material for more examples.

a slight overlap. In line with this, SimSaC could integrate the VPR pipeline for end-to-end learning. The integration would allow a single neural network system for various applications such as SLAM and visual surveillance.

Conclusion. The proposed SimSaC—which leverages both dense correspondence and mis-correspondence—solves performance degradation with imperfect matches from real-world scenarios that conventional change detection algorithms display a huge performance drop. For training SimSaC without additional annotations, we carefully designed the training scheme with geometric transformations and the cut-paste method. We verified that the proposed training scheme advances the robustness of change detection even given imperfect matches. Furthermore, we collected a test benchmark dataset for evaluating change detection algorithms placed within real-world applications. We expect our work would stimulate the progress of the change detection task towards practical use cases.

Acknowledgement. This work was supported in part by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by Korean Government (MSIT) (No.2020-0-00440, Development of Artificial Intelligence Technology that Continuously Improves Itself as the Situation Changes in the Real World) and in part by the IITP grant funded by MSIT (No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST)).

References

[1] Thangarajah Akilan, Qingming Jonathan Wu, Amin Safaei, Jie Huo, and Yimin Yang. A 3d cnn-lstm-based image-to-image foreground segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 21(3):959–971, 2020. 2

[2] Pablo F Alcantarilla, Simon Stent, German Ros, Roberto Arroyo, and Riccardo Gherardi. Street-view change detection with deconvolutional networks. *Autonomous Robots*, 42(7):1301–1322, 2018. 2, 5

[3] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5297–5307, 2016. 3

[4] Relja Arandjelovic and Andrew Zisserman. All about vlad. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1578–1585, 2013. 3

[5] Mohammed Chafik Bakkay, Hatem A Rashwan, Houssam Salmane, Louahdi Khoudour, D Puig, and Yassine Ruichek. Bscgan: Deep background subtraction with conditional generative adversarial networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4018–4022. IEEE, 2018. 2

[6] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, pages 404–417. Springer, 2006. 3

[7] Francesca Bovolo. A multilevel parcel-based approach to change detection in very high resolution multitemporal images. *IEEE Geoscience and Remote Sensing Letters*, 6(1):33–37, 2008. 2

[8] Alexander Buslaev, Vladimir I Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A Kalinin. Albumentations: fast and flexible image augmentations. *Information*, 11(2):125, 2020. 6

[9] Turgay Celik and Kai-Kuang Ma. Unsupervised change detection for satellite images using dual-tree complex wavelet transform. *IEEE Transactions on Geoscience and Remote Sensing*, 48(3):1199–1210, 2009. 2

[10] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009. 1

[11] Shuo Chen, Kailun Yang, and Rainer Stiefelhagen. Dr-tanet: Dynamic receptive temporal attention network for street scene change detection. In *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2021. 6, 7

[12] Yingying Chen, Jinqiao Wang, Bingke Zhu, Ming Tang, and Hanqing Lu. Pixelwise deep sequence learning for moving object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9):2567–2579, 2017. 2

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 5

[14] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, volume 1, pages 1–2. Prague, 2004. 3

[15] Rodrigo Caye Daudt, Bertr Le Saux, and Alexandre Boulch. Fully convolutional siamese networks for change detection. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4063–4067. IEEE, 2018. 6, 7

[16] JS Deng, K Wang, YH Deng, and GJ Qi. Pca-based land-use change detection and analysis using multitemporal and multisensor satellite data. *International Journal of Remote Sensing*, 29(16):4823–4838, 2008. 2

[17] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 224–236, 2018. 3

[18] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766, 2015. 2

[19] Debidatta Dwibedi, Ishan Misra, and Martial Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV),*, pages 1301–1310, 2017. 2, 4

[20] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010. 6

[21] Enqiang Guo, Xinsha Fu, Jiawei Zhu, Min Deng, Yu Liu, Qing Zhu, and Haifeng Li. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. *arXiv preprint arXiv:1810.09111*, 2018. 6, 7

[22] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14141–14152, 2021. 3, 5

[23] Bin Hou, Qingjie Liu, Heng Wang, and Yunhong Wang. From w-net to cdgan: Bitemporal change detection via deep learning techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 58(3):1790–1802, 2020. 2

[24] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8981–8989, 2018. 2

[25] Chunlei Huo, Zhixin Zhou, Hanqing Lu, Chunhong Pan, and Keming Chen. Fast object-level change detection for vhr images. *IEEE Geoscience and Remote Sensing Letters*, 7(1):118–122, 2010. 2

[26] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3277–3285, 2017. 5

[27] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2462–2470, 2017. 2, 3, 4

[28] Jungho Im, JR Jensen, and JA Tullis. Object-based change detection using correlation image analysis and image segmentation. *International journal of remote sensing*, 29(2):399–423, 2008. 2

[29] Tommi S Jaakkola, David Haussler, et al. Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems*, pages 487–493, 1999. 3

[30] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision*, 12(1–3):1–308, 2020. 1

[31] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3304–3311. IEEE, 2010. 3

[32] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 4(3), 2013. 5

[33] In Su Kim, Hong Seok Choi, Kwang Moo Yi, Jin Young Choi, and Seong G Kong. Intelligent visual surveillance—a survey. *International Journal of Control, Automation and Systems*, 8(5):926–939, 2010. 1

[34] Mathieu Labbé and François Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2):416–446, 2019. 5

[35] Yinjie Lei, Duo Peng, Pingping Zhang, Qiuhong Ke, and Haifeng Li. Hierarchical paired channel fusion network for street scene change detection. *IEEE Transactions on Image Processing*, 30:55–67, 2020. 6, 7

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755. Springer, 2014. 5

[37] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[38] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157. IEEE, 1999. 3

[39] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2015. 1

[40] Murari Mandal, Vansh Dhar, Abhishek Mishra, Santosh Kumar Vipparthi, and Mohamed Abdel-Mottaleb. 3dcd: Scene independent end-to-end spatiotemporal feature learning framework for change detection in unseen videos. *IEEE Transactions on Image Processing*, 30:546–558, 2021. 2

[41] Iaroslav Melekhov, Aleksei Tiulpin, Torsten Sattler, Marc Pollefeys, Esa Rahtu, and Juho Kannala. Dgc-net: Dense geometric correspondence network. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1034–1042. IEEE, 2019. 2, 3, 4

[42] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2017. 3

[43] Ulrich Nehmzow. *Mobile robotics: a practical introduction*. Springer Science & Business Media, 2012. 1

[44] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4990–4999, 2017. 6

[45] Tien Phuoc Nguyen, Cuong Cao Pham, Synh Viet-Uyen Ha, and Jae Wook Jeon. Change detection by training a triplet network for motion feature extraction. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(2):433–446, 2019. 2

[46] Allan Aasbjerg Nielsen. The regularized iteratively reweighted mad method for change detection in multi-and hyperspectral data. *IEEE Transactions on Image processing*, 16(2):463–478, 2007. 2

[47] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3456–3465, 2017. 3

[48] Eng-Jon Ong, Syed Sameed Husain, Mikel Bober-Irizar, and Miroslaw Bober. Deep architectures and ensembles for semantic video classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(12):3568–3582, 2018. 3

[49] Jin-Man Park, Jae-Hyuk Jang, Sahng-Min Yoo, Sun-Kyung Lee, Ue-Hwan Kim, and Jong-Hwan Kim. Changesim: Towards end-to-end online scene change detection in industrial indoor environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021. 5

[50] Florent Perronnin and Christopher Dance. Fisher kernels on visual vocabularies for image categorization. In *2007 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2007. 3

[51] K Ram Prabhakar, Akshaya Ramaswamy, Suvaansh Bhambri, Jayavardhana Gubbi, R Venkatesh Babu, and Balamuralidhar Purushothaman. Cdnet++: Improved change detection with deep neural network feature correlation. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020. 6, 7

[52] Richard J Radke, Srinivas Andra, Omar Al-Kofahi, and Badrinath Roysam. Image change detection algorithms: a

systematic survey. *IEEE transactions on image processing*, 14(3):294–307, 2005. 1

[53] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4161–4170, 2017. 2

[54] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelovic, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Ncnet: Neighbourhood consensus networks for estimating image correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 3

[55] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571. IEEE, 2011. 3

[56] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. In *The British Machine Vision Conference (BMVC)*, volume 61, pages 1–12, 2015. 2, 5

[57] Ken Sakurada, Mikiya Shibuya, and Weimin Wang. Weakly supervised silhouette-based semantic scene change detection. In *2020 IEEE International conference on robotics and automation (ICRA)*, pages 6861–6867. IEEE, 2020. 5, 6, 7

[58] Ken Sakurada, Weimin Wang, Nobuo Kawaguchi, and Ryosuke Nakamura. Dense optical flow based change detection network robust to difference of camera viewpoints. *arXiv preprint arXiv:1712.02941*, 2017. 6, 7

[59] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, volume 3, pages 1470–1470. IEEE Computer Society, 2003. 3

[60] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8934–8943, 2018. 2

[61] Prune Truong, Martin Danelljan, and Radu Timofte. Glunet: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6258–6268, 2020. 3

[62] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision (ECCV)*, pages 467–483. Springer, 2016. 3

[63] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 633–641, 2017. 5