Learning to Detect Fine-Grained Change under Variant Imaging Conditions

Rui Huang[1,2,4], Wei Feng[1,4]*, Zezheng Wang[1,4], Mingyuan Fan[1,4], Liang Wan[3,4], Jizhou Sun[1,4]
[1] School of Computer Science and Technology, Tianjin University, Tianjin, China
[2] School of Computer Science and Technology, Civil Aviation University of China, Tianjin, China
[3] School of Computer Software, Tianjin University, Tianjin, China
[4] Key Research Center for Surface Monitoring and Analysis of Cultural Relics, SACH, China
{ruihuang, wfeng, zzwang, myfan, lwan, jzsun}@tju.edu.cn

Abstract

Fine-grained change detection under variant imaging conditions is an important and challenging task for high-value scene monitoring in culture heritage. State-of-the-art methods solve this problem by jointly optimizing three related factors, i.e., camera pose difference, illumination variation, and the true minute change of the scene. Their performances are highly dependent on the delicate choice of key parameters, which significantly limits their feasibility in real-world applications. In this paper, we show that after a simple coarse alignment of lighting and camera differences, fine-grained change detection can be reliably solved by a deep network model, which is specifically composed of three functional parts, i.e., camera pose correction network (PCN), fine-grained change detection network (FCDN), and detection confidence boosting. Since our model is properly pre-trained and fine-tuned on both general and specialized data, it exhibits very good generalization capability to produce high-quality minute change detection on real-world scenes under varied imaging conditions. Extensive experiments validate the superior effectiveness and reliability over state-of-the-art methods. We have achieved 67.41% *relative F1-measure improvement over the best competitor on real-world benchmark dataset.*

1. Introduction

Change detection is a widely studied problem that is broadly useful in a lot of computer vision applications, such as scene abnormal detection, visual surveillance, remote sensing, vision based automatic driving [9, 10, 17, 18, 41]. Most studies about change detection focus on extracting large-scale significant changes of the scene with relatively constant illuminations and fixed camera poses [31, 37].

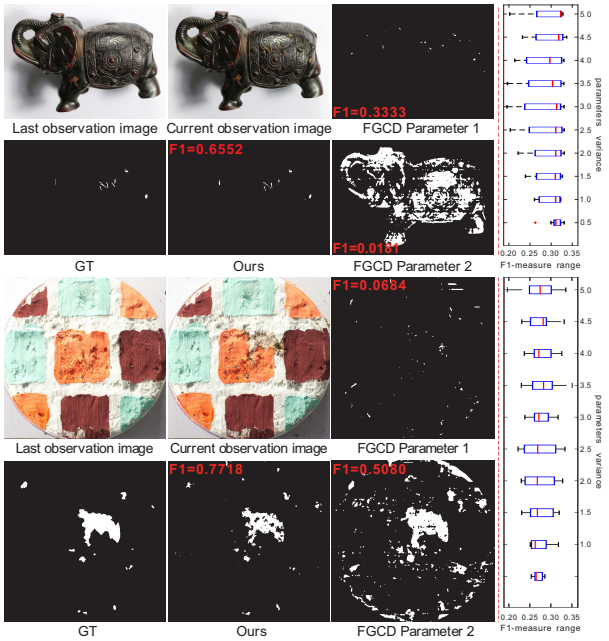

Figure 1. Generalization power and parameters robustness of different fine-grained change detectors. For each testing example, we conduct two kinds of experiments, whose results are separated by a red dotted line. The left side respectively shows the two observation images, ground truth (GT), minute change detection results of our approach (Ours) and FGCD detector [12] using two different parameters (Parameter 1 & 2). We can see for the 1st case, parameter 1 produces more promising detection than parameter 2, while parameter 2 is much better than parameter 1 for the 2nd case. In contrast, the proposed DeepFCD exhibits very good generalization ability and can constantly generate reliable minute change detection results for both cases. On the right side, we show the influence of parameters variances to the minute change detection accuracy (i.e., F1-measure), from which we can clearly see the sensitivity of state-of-the-art FGCD detector [12] to key parameters.

Recently, a new problem, fine-grained change detection [12], has been proposed to discover and localize the minute changes occurred in real-world high-value scenes, e.g., ancient murals, sculptures, inscriptions and allimpor-

*Corresponding author. Tel:(+86)-22-27406538. This work is supported by NSFC 61671325, 61572354, 61672376.

tant buildings. This problem could be very challenging, since the imaging conditions, i.e., the lighting conditions and camera poses, are not fixed, but the target changes are subtle and fine-grained. State-of-the-art methods [12, 34] solve fine-grained change detection as a joint optimization problem of the three most related factors, i.e., camera pose difference, illumination variation, and the true minute change of the scene. However, as shown in Fig. 1, the performance of such explicit joint optimization strategy via low-level energy minimization [12] highly depends on the delicate choice of key parameters. It is hard, if only possible, for such methods to find an optimal parameters setting to be universally applicable to most cases. Hence, it is highly necessary to provide a nonparametric fine-grained change detector with much better generalization power.

In this paper, we try to import the great generalization power of deep learning into this new and challenging problem. An inevitable obstacle for the proposed DeepFCD model is the severe lacking of large-volume real-world scenes with fine-grained changes. Hence, besides the FGCD benchmark [12], we elaborately captured 305 groups of real-world scenes with artificial or natural minute changes. To further boost the number of training samples, we present *scene-aware minute change augmentation* to produce enough amount of effective training samples from the selected training image dataset. We find that, after a simple coarse alignment of illumination and camera pose differences, fine-grained change detection can be satisfactorily solved a hybrid deep network model, namely DeepFCD. Specifically, our model is composed of three functional components: 1) camera *pose correction network* (PCN), a dual model formed by a 4-layer CNN and a classical optical flow generator, e.g., SiftFlow [26], 2) *fine-grained change detection network* (FCDN), a 9-layer CNN that takes stacked multi-lighting images of two observations as input and directly outputs minute change map, and 3) *detection confidence boosting* (BS). We propose a *spatial alignment layer* to connect PCN and FCDN, and develop the corresponding error backpropagation procedure to realize fine-tuning of the whole network. Our model is firstly pre-trained on general data, followed by fine-tuning on the specialized FGCD augmentation dataset. Extensive experiments validate the superior generalization power and much better performance over state-of-the-art fine-grained change detectors. Our model particularly achieves 67.41% relative F1-measure improvement, with 0.242 absolute F1-measure boosting, compared to the best competitor, FGCD-S [12] (with overall F1-measure 0.359) on real-world testing set.

2. Related Work

Scene change detection. Although a lot of successful approaches have been proposed to handle general change detection for large-scale application, e.g., remote sensing [18, 41], urban environment monitoring [9, 31, 36, 37], such methods are almost infeasible for fine-grained change detection, due to its high precision requirement and the property being sensitive to illumination and camera variation. Recently, several researchers [12, 34] have proposed solutions for fine-grained change detection. For example, Feng et al. [12] propose a jointly optimized geometry correction, lighting correction and change detection method and provide a benchmark dataset for minute change detection. Stent et al. [34] present a minute change detection method with absolute difference based on elaborate pose and lighting correction. Although such methods are able to generate good results, they have to select key parameters carefully for different scenes, which is not practical for real-world application that may contain various noise and complex scene changes. Unlike the above low-level vision based methods, we propose a fully deep network based solution for fine-grained change detection, which helps to get rid of parameter selection while obtains significant detection precision improvement.

Image alignment and dense correspondence. Dense correspondence via optical flow is significantly important for fine-grained change detection [12]. Recent work [11] proposes to control the camera pose before shooting image. However, more works use optical flow to conduct image alignment. Traditional optical flow methods using low-level feature and optimization framework [1, 35] are able to generate satisfied results for less complicated scenes. However, when illumination and camera variation become serious, such methods usually fail. Although deep learning based optical flow has been proposed, i.e., the FlowNet [8], it can only estimate coarse corresponds and easily influenced by illumination. Therefore, all above methods are not available for fine-grained change detection when considering complex illumination and camera variations. Instead of naively training an end-to-end CNN for optical flow, we present a pose correction network (PCN), which consists of dual model formed by a 4-layer CNN and SiftFlow [26]. CNN model leans incremental quantity to compensate SiftFlow [26] for precise pose correction. The experiment results show that PCN can boost the change detection performance.

Illumination correction. Illumination correction also plays an important role in fine-grained change detection. Although a lot of methods including lighting correction and color constancy [2, 14, 15, 38, 40], intrinsic image decomposition [3, 6, 21, 29] and even the deep learning based methods [4, 13] have been proposed recently and can be used to correct illumination, such methods still cannot directly apply to precisely end-to-end fine-grained change detection. In [12], the authors propose a simple lighting correction in an iterative optimization fine-grained change detection framework, wherein, lighting parameters needed in-

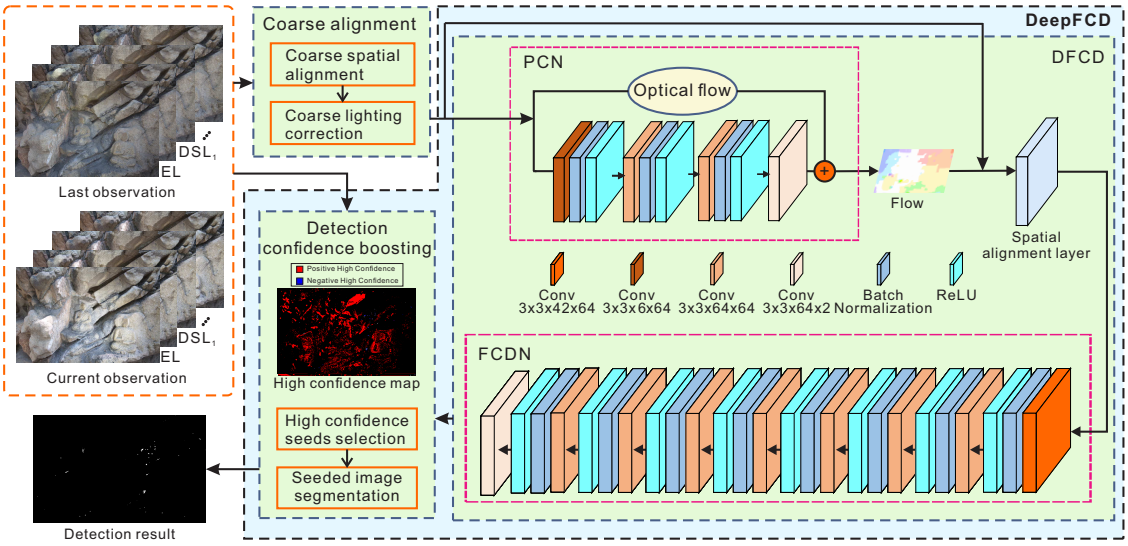

Figure 2. Framework of the proposed deep fine-grained change detector, DeepFCD. See text for details.

tentional adjustment for different scenes. In this paper, our DeepFCD tolerates the deficiency of using coarse lighting correction method [12] with same parameters on all scenes and obtains much speed boost.

Deep learning and applications. Since the success of the ImageNet [25], deep learning has obtained significant progressive in classification, recognition, detection [7, 16, 22, 24, 27, 30, 32]. However, the requirement of large-volume training data becomes bottleneck of applying deep learning to specific task, e.g., fine-grained change detection. Recent work [23] tries to use a weakly supervised approach to train a CNN for change detection. However, they ignore the pose and illumination differences between two observations which results in coarse change detection. For fine-grained change detection, there are not enough training data, even lack of real changes in real-world scenes. In this paper, we propose feasible scene-aware minute change augmentation to guarantee the amount of effective training samples. In addition, we propose a spatial alignment layer to connect PCN with FCDN, and develop the corresponding error backpropagation procedure to realize fine-tuning of the whole network.

3. Learn to Detect Fine-Grained Changes

As illustrated in Fig. 2, given the last observations $\mathrm{X} = [\mathrm{x}_{\mathrm{EL}}, \mathrm{x}_{\mathrm{DSL}_1}, \cdots, \mathrm{x}_{\mathrm{DSL}_k}]$ and the current observations $\mathrm{Y} = [\mathrm{y}_{\mathrm{EL}}, \mathrm{y}_{\mathrm{DSL}_1}, \cdots, \mathrm{y}_{\mathrm{DSL}_k}]$, our goal is detecting the changes from X to Y. Images with subscript EL are collected under environment lighting, while images with subscript $\mathrm{DSL_i}$ are collected under the i-th directional side lighting. The camera pose and lighting conditions may be different when collecting the two observations. The whole process of fine-grained change detection can be modeled by

$$C = F(\mathrm{X}, \mathrm{Y}) = F'(\mathrm{X}', \mathrm{Y}'), \tag{1}$$

where X' and Y' are coarse aligned X and Y, F' is the change detection function with coarse aligned X' and Y'. In this paper, we model F' by a deep convolutional neural network, DFCD, which consists of two subnetworks PCN and FCDN. PCN is devised for fine scale pose correction which can eliminate the effects of the unify parameters in coarse alignment. FCDN is for change detection, which can tolerate misalignment of pose and lighting between X and Y. Thus Eq. (1) can be rewritten as

$$C = F'(\mathrm{X}', \mathrm{Y}'; \mathrm{W}_{\mathrm{PCN}}, \mathrm{W}_{\mathrm{FCDN}}), \tag{2}$$

where $\mathrm{W}_{\mathrm{PCN}}$ and $\mathrm{W}_{\mathrm{FCDN}}$ are the parameters of PCN and FCDN, respectively. In this paper, we first pre-train PCN and FCDN independently. And then we joint tune them by a novel layer, i.e., spatial alignment layer.

For coarse alignment, we first conduct white balance operation on all observed images to remove the color of illuminations. This is different from previous work [12] that applies global photometric correction to the previous observations X. We actually find that the white balance operation can remove the illuminations like global photometric correction, but takes much less time. Next, coarse spatial alignment is achieved by SiftFlow [26]. We then employ the normal-aware lighting correction [12] for coarse lighting correction. All parameters of the coarse alignment are fixed in our evaluations. In this paper, we use MatConvNet [39] for network training and testing.

3.1. Scene-aware minute change augmentation

The amount of fine-grained change data of real-world scenes is small for training. To effectively train our deep

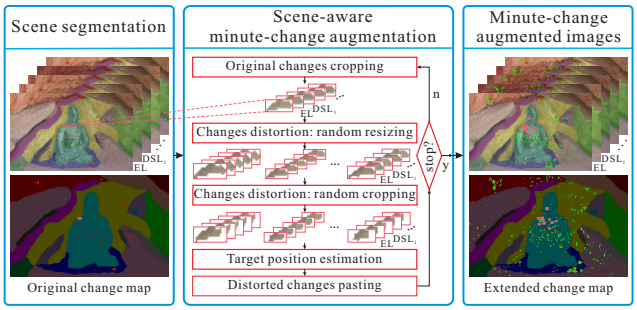

Figure 3. Scene-aware minute change augmentation. Green indicates augmented changes, and red denotes original changes.

model with abundant data, we synthesize more data by pasting spatially transformed changes on images. We paste the regions of changes to the regions of other scenes that have similar material, texture and appearance. This scene-aware minute change augmentation is shown in Fig. 3.

We first conduct scene segmentation on current observations $\mathrm{Y} = [\mathrm{y}_{\mathrm{EL}}, \mathrm{y}_{\mathrm{DSL}_1}, \cdots, \mathrm{y}_{\mathrm{DSL}_k}]$ according to the material, texture and appearance of scene. Note that this can be conducted by automatical segmentation algorithms or by human. In our experiment, our staffs label each scenes by hand to guarantee the segmentation quality. Then we augment the changes by randomly cropping and resizing each real change and paste them to the positions where similar changes have high probability to occur. Let $\mathbb{C} = \{\mathrm{c}^1, \mathrm{c}^2, \ldots, \mathrm{c}^{\mathrm{N}}\}$ denotes the original change region set, where $\mathrm{c}^{\mathrm{j}} = [\mathrm{c}^{\mathrm{j}}_{\mathrm{EL}}, \mathrm{c}^{\mathrm{j}}_{\mathrm{DSL}_1}, \cdots, \mathrm{c}^{\mathrm{j}}_{\mathrm{DSL}_k}]$ is the real change region under environment and directional side lightings. Specially, for each scene, we crop each original change corresponding the multiple lightings and distort it with random resizing and cropping. The resize ratio is randomly selected from $[\frac{4}{8}, \frac{5}{8}, \ldots \frac{12}{8}]$. The areas of the cropped changes are more than half of the area of original change. The resizing and cropping process can be formed by $\widehat{\mathrm{c}}^{\mathrm{j}} = f(\mathrm{c}^{\mathrm{j}})$. We estimate the paste positions of the distorted change $\widehat{\mathrm{c}}^{\mathrm{j}}$ by selecting the closest position that randomly sampled K ($K = 10$) positions from the same segment. Note that we pasted the distorted changes to the corresponding lighted images. By pasting the distorted changes to the current observations, we can use a sliding window to uniformly collecting training samples. We discard the sample that total changes' area is less then 0.8 times of a sample's area. Here, the change's area is calculated as the area of its bounding box.

3.2. Learning of camera pose correction

The purpose of camera pose correction network PCN is to eliminate the effects of using unify parameters in coarse alignment and achieve fine scale pose correction. However, we do not need to design a complex network like FlowNet [8] for accurate optical flow estimation. A cheap way is using traditional optical flow method (e.g., SiftFlow [26]) and computing a compensation quantity for precise optical flow. In this paper, we use a four layer convolutional neural network for computing the compensation quantity. Thus we can formalize our PCN by

$$\begin{aligned} f_{\mathrm{PCN}} =& \mathrm{F}_{\mathrm{PCN}}(\mathrm{X}', \mathrm{Y}'; \mathrm{W}_{\mathrm{PCN}}) \\ =& \mathrm{F}_{\mathrm{SiftFlow}}(\mathrm{X}', \mathrm{Y}') + \mathrm{F}_{\mathrm{CNN}}(\mathrm{X}', \mathrm{Y}'; \mathrm{W}_{\mathrm{CNN}}), \\ =& f_{\mathrm{SiftFlow}} + f_{\mathrm{CNN}} \end{aligned} \quad (3)$$

where f_{SiftFlow} is optical flow that computed by SiftFlow [26] function $\mathrm{F}_{\mathrm{SiftFlow}}$, f_{CNN} is compensation quantity that computed by the proposed four layer CNN function $\mathrm{F}_{\mathrm{CNN}}$. Actually $\mathrm{W}_{\mathrm{PCN}}$ is identical to $\mathrm{W}_{\mathrm{CNN}}$. Unlike FlowNet [8], which needs lots of training data to train, PCN uses a side branch for computing the residual of optical flow between the ground truth with SiftFlow [26], which is more easy to learn. We learn PCN by minimizing

$$\|f_{\mathrm{GT}} - f_{\mathrm{PCN}}\|_2 + \lambda \|f_{\mathrm{PCN}}\|_2, \quad (4)$$

where f_{GT} is ground truth optical flow, and f_{PCN} is the estimated optical flow via the dual-model.

Architecture. As shown in Fig. 2, our camera pose correction network PCN is a dual model of two components. One component is a conventional optical flow detector, for which we use SiftFlow [26] due to its efficiency. The output of the first component is a coarsely estimated optical flow field. The second component is a 4-layer convolutional neural networks. It concatenates the pair of images as the input and the output compensates the quantity of coarse flow estimation, which is added to the coarse optical flow estimates to get better estimation. Note that we partition layers by convolution operations and therefore, batch normalization and ReLU do not belong to individual layers. We use 3×3 filters, and pad the input images with one all-0 row/column out of each side of the image. The last convolutional layer has two outputs corresponding to the flow estimation of two images. The dimension of the output feature is set to 64 in the whole net. Besides the fourth layer, all of the previous three layers use batch normalization and ReLU.

Pre-training procedure. PCN is pre-trained on the general MPI Sintel Flow dataset with 1041 training image pairs [5]. We randomly initialize the parameters of PCN and train it by stochastic gradient descent algorithm. We use fixed learning rate 10^{-11}, batch size 1, weight decay (i.e., λ) 0.0005, and momentum 0.95. We have run 17 epoches. To use PCN, we take pair of images with corresponding lighting as input. The average of estimated optical flows under all lightings is calculated as final optical flow.

3.3. Deep fine-grained change detection

After the coarse alignment and fine scale pose correction by PCN, we use a nine layers network for minute change detection. This is different from the previous work FGCD [12] that uses low-rank model to iteratively compute the minute

changes, our model directly computes the minute changes with deep model that has more powerful recognition and generalization abilities. The fine-grained change detection network FCDN learns to detect minute changes from large real-world scenes. It can tolerate the diversity of the real-world minute changes and can tolerate the effects of the unify of the parameters. We formalize this process by

$$\mathrm{C} = \mathrm{F}_{\mathrm{FCDN}}(\mathrm{X}'', \mathrm{Y}'; \mathrm{W}_{\mathrm{FCDN}}), \tag{5}$$

where X'' is the pose corrected X', $\mathrm{W}_{\mathrm{FCDN}}$ is the parameters of FCDN.

Architecture. Our fine-grained change detection network FCDN is shown at the bottom right of Fig.2. FCDN takes a $H \times W \times 42$ matrix as input, which consists of previous and current observations under environment lighting and 6 directional side lightings, arranged in pairs, i.e., $[\mathrm{X}_{\mathrm{EL}}; \mathrm{Y}_{\mathrm{EL}}; \mathrm{X}_{\mathrm{DSL}_1}; \mathrm{Y}_{\mathrm{DSL}_2}; \cdots; \mathrm{X}_{\mathrm{DSL}_6}; \mathrm{Y}_{\mathrm{DSL}_6}]$. The output of FCDN is a $H \times W \times 2$ matrix, where one channel is the unchange probability map and the other is the change probability map. The entire network employs 9 layers of building blocks with convolution, batch normalization and ReLU. The output feature numbers in middle layers are also set to 64. Each convolutional layer uses filters of size 3×3. When performing convolution, we pad one all-0 row/column to each side of the image to maintain the spatial resolution.

Pre-training procedure. To effectively train FCDN, we first do coarse alignment for all training samples, which are then sequentially processed by the normal-aware lighting correction and camera pose correction procedure of state-of-the-art minute change detector FGCD [12]. Note, all key parameters of FGCD are fixed to generate the pre-training samples of FCDN. The parameters of FCDN are randomly initialized, and the network is trained by stochastic gradient descent. We adopt multinomial logistic loss for training and use fixed learning rate 10^{-3}, batch size 50, weight decay 0.0005, and momentum 0.95. We have run 20 epoches.

3.4. Jointly fine-tuning FCDN and PCN

Although pre-trained PCN and FCDN can be connected to detect fine-grained changes, we show that it is easy to learn DFCD that can jointly fine-tune PCN and FCDN, as shown in the right of Fig. 2. To back propagate the derivatives of FCDN to PCN, we remove the loss function of PCN and add a spatial alignment layer between FCDN and PCN. Spatial alignment layer takes coarsely aligned images and optical flow as inputs and output spatially aligned images. Formally, F_{SA} denotes forward computation of spatial alignment layer, and $\mathrm{F}_{\mathrm{FCDN}}$ denotes fine-grained change detection function. The spatial alignment layer is used to calculate pose corrected observation X_{SA} as

$$\mathrm{X}_{\mathrm{SA}} = \mathrm{F}_{\mathrm{SA}}(\mathrm{X}', f_{\mathrm{PCN}}), \tag{6}$$

Next we concatenate X_{SA} and Y' to compute changes by

$$\mathrm{C} = \mathrm{F}_{\mathrm{FCDN}}(\mathrm{X}_{\mathrm{SA}}, \mathrm{Y}'; \mathrm{W}_{\mathrm{FCDN}}), \tag{7}$$

In general optical flow algorithm, F_{SA} is a warping function. Nevertheless, it is difficult to compute the derivative of F_{SA}. To facilitate the derivative computation, we approximate Eq. (6) by $\mathrm{X}_{\mathrm{SA}}(\mathrm{p}) = \mathrm{X}'(p')$, where $p' = p + f_{\mathrm{PCN}}$. Note that p and p' are pixel positions. Thus, the f_{PCN} derivative of loss L can be calculated by

$$\begin{aligned}
\frac{\partial L}{\partial f_{\mathrm{PCN}}} &= \frac{\partial L}{\partial \mathrm{F}_{\mathrm{FCDN}}} \frac{\partial \mathrm{F}_{\mathrm{FCDN}}}{\partial \mathrm{X}_{\mathrm{SA}}} \frac{\partial \mathrm{X}_{\mathrm{SA}}}{\partial f_{\mathrm{PCN}}} \\
&= \frac{\partial L}{\partial \mathrm{F}_{\mathrm{FCDN}}} \frac{\partial \mathrm{F}_{\mathrm{FCDN}}}{\partial \mathrm{X}_{\mathrm{SA}}} \frac{\partial \mathrm{X}_{\mathrm{SA}}}{\partial p} \frac{\partial p}{\partial f_{\mathrm{PCN}}} \\
&= \frac{\partial L}{\partial \mathrm{F}_{\mathrm{FCDN}}} \frac{\partial \mathrm{F}_{\mathrm{FCDN}}}{\partial \mathrm{X}_{\mathrm{SA}}} \frac{\partial \mathrm{X}_{\mathrm{SA}}}{\partial p}(-1),
\end{aligned} \tag{8}$$

where $\frac{\partial \mathrm{X}_{\mathrm{SA}}}{\partial p}$ is image gradient on horizontal and vertical directions. After getting the derivative of f_{PCN}, we can compute derivatives of parameters in each layer by standard backpropagation.

The DFCD network is fine-tuned by stochastic gradient descent with multinomial logistic loss. We use fixed learning rate 10^{-6} for FCDN and 10^{-11} for PCN, batch size 15, weight decay 0.0005, and momentum 0.95. We use all our augmented training samples, which are first processed by our coarse alignment in this fine-tuning stage.

3.5. Detection confidence boosting

For each pixel p, we measure its detection confidence by

$$\mathrm{S}_{\mathrm{conf}}(p) = 1 - e^{\min(\mathrm{P}_{\mathrm{P}}(p), \mathrm{P}_{\mathrm{N}}(p)) - \max(\mathrm{P}_{\mathrm{P}}(p), \mathrm{P}_{\mathrm{N}}(p))}, \tag{9}$$

where $\mathrm{P}_{\mathrm{P}}(p)$ and $\mathrm{P}_{\mathrm{N}}(p)$ are the outputs of FCDN indicating the positive and negative possibility of pixel p, respectively. Note, larger $\mathrm{S}_{\mathrm{conf}}(p)$ means higher fine-grained change detection confidence. From $\mathrm{S}_{\mathrm{conf}}$, we can derive a high confidence detection map S_{hc} by thresholding $\mathrm{S}_{\mathrm{hc}}(p) = \mathbf{1}(\mathrm{S}_{\mathrm{conf}}(p) \geq \alpha)$, where $\mathbf{1}(\cdot)$ is a boolean function with $\mathbf{1}(true) = 1$ and $\mathbf{1}(false) = 0$.

As shown in the top row of Fig. 4, the precisions of our positive and negative high confidence detections are both very high, but the recall of our positive detection is not good enough. Hence, it is necessary to propagate the correct positive detections in the image domain. To this end, we propose to use high confidence positive and negative detection pixels as foreground and background seeds, respectively, and conduct seeded image segmentation [20] to further boost the detection accuracy of our model. As validated by the bottom row of Fig. 4, by setting $\alpha = 0.934$, the proposed detection confidence boosting can help to further improve the F1-measure by 0.016.

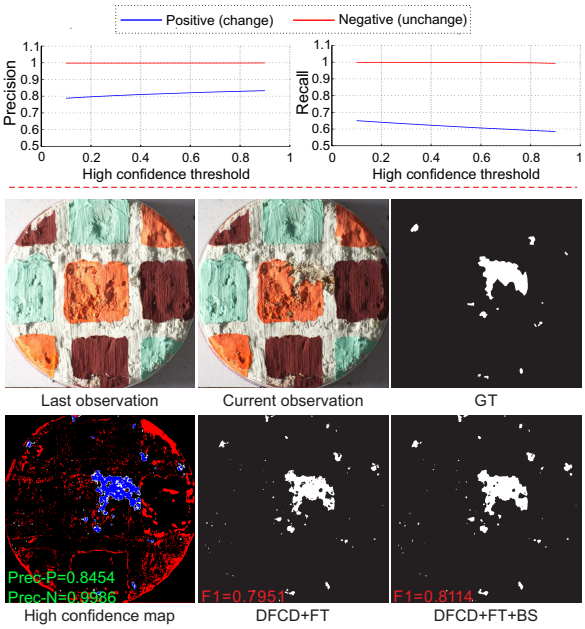

Figure 4. Detection confidence boosting. Top row: Average precisions of pixel-level high confidence positive and negative detections on the whole training set using different threshold α. Bottom row: An example of high confidence detection map $\mathrm{S}_{\mathrm{conf}}$, where positive and negative detections are show in blue and red colors, respectively. Note, the precision of high confidence positive and negative detections are 0.8454 and 0.9986, respectively.

4. Experimental Results

4.1. Setup

Dataset. In [12], a real-world dataset was built for fine-grained change detection of misaligned scenes under variant imaging conditions, denoted as FGCD. The FGCD dataset consists of 16 scenes belonging to three subsets, i.e., natural changes in outdoor scenes ($\mathbf{D_p}$), real changes under laboratory conditions ($\mathbf{D_b}$) and artificial changes on statues ($\mathbf{D_s}$). In this paper, we built a new subset ($\mathbf{D_{ext}}$), containing 12 outdoor scenes and 293 indoor scenes. For each scene, we capture 7 images under different illumination conditions, including one environment lighting and 6 directional side lightings. Combining the four subsets together forms our dataset $\mathrm{FGCD}_{\mathrm{ext}}$, with more details listed in Table 1.

Table 1. Details of datasets.

Dataset	$\mathbf{D_p}$	$\mathbf{D_b}$	$\mathbf{D_s}$	$\mathbf{D_{ext}}$	Total
$\mathrm{FGCD}_{\mathrm{ext}}$	2	10	4	305	321
$\mathrm{FGCD}_{\mathrm{ext-Train}}$	1	8	3	291	303
$\mathrm{FGCD}_{\mathrm{ext-Test}}$	1	2	1	14	18
$\mathrm{FGCD}_{\mathrm{ext-Aug}}$	42	288	3	510	843
$\mathrm{FGCD}_{\mathrm{ext-AugSp}}$	2100	17244	295	27491	44314

The testing dataset $\mathrm{FGCD}_{\mathrm{ext-Test}}$ is formed by selecting 18 scenes randomly from the four subsets in $\mathrm{FGCD}_{\mathrm{ext}}$. To train our FCDN networks, we first conduct scene-aware minute change augmentation (Sec. 3.1) to $\mathrm{FGCD}_{\mathrm{ext-Train}}$, to obtain an augmented training dataset $\mathrm{FGCD}_{\mathrm{ext-Aug}}$, containing 843 groups of images. The training sample set $\mathrm{FGCD}_{\mathrm{ext-AugSp}}$ is finally generated by sampling $44,314$ groups of image patches from $\mathrm{FGCD}_{\mathrm{ext-Aug}}$, with a patch size of 128×128. From $\mathrm{FGCD}_{\mathrm{ext-AugSp}}$, we randomly select 90% samples for training, and the remaining 10% are used for validation.

Baselines. We select the state-of-the-art fine-grained change detection method FGCD [12] and two change detection methods, SC_SOBS [28] and SENSE [33], which reported best results in CDNet challenge 2012 and 2014 [19], as baseline methods.

FGCD [12] uses two strategies for change decision, i.e. linear SVM (denoted as FGCD-S) and simple difference plus threshold (denoted as FGCD-T). We use the training dataset $\mathrm{FGCD}_{\mathrm{ext-Train}}$ to train the SVM model for FGCD-S, and find an optimal threshold via difference evolution for FGCD-T. We then run their codes with recommended parameters on the testing dataset $\mathrm{FGCD}_{\mathrm{ext-Test}}$.

SC_SOBS [28] and SENSE [33] require a pair of two images as inputs. Since our image data for one scene contain multiple images, we apply two strategies to generate SOBS and SENSE results. The first strategy is to use every two images as the input and average all detection results (e.g. denoted as SC_SOBS M). The second strategy is to average all the images before change detection (e.g. denoted as SC_SOBS A). In addition, we conduct our lighting correction and pose correction as preprocessing for SC_SOBS and SENSE. These method variants are denoted with a mark LF (e.g., SENSE LF).

In total, we have 10 baseline methods. Except for FGCD variants, we tune each method via difference evolution by evaluating F1-measure on the training set. We also test 4 variants of our method. To be specific, FCDN denotes the minute changes using SiftFlow only; DFCD denotes the minute changes using PCN; DFCD+FT denotes DFCD with fine-tuning; and DFCD+FT+BS denotes DFCD with fine-tuning and detection confidence boosting.

Criteria. Following the experiment settings of [19], we use F1-measure (F1), precision (Pr) and recall (Re) for quantitative evaluation.

4.2. Quantitative comparison

The quantitative evaluation results of different methods on $\mathbf{D_{ext}}$, $\mathbf{D_p}$, $\mathbf{D_b}$, $\mathbf{D_s}$ are reported in Table 2. From the table, we find that our preprocessing is helpful to improve SC_SOBS [28] and SENSE [33]. However, the improved variants still get poor performance.

From Table 2 for dataset $\mathbf{D_{ext}}$, we find that fine-grained change detection methods are consistently better than conventional change detection methods in terms of F1-measure. We also note that FGCD-S is better than FGCD-T, as presented in [12]. Our approach DFCD+FT+BS can sig-

Table 2. Average F1-measure of compared methods on datasets $\mathbf{D_{ext}}$, $\mathbf{D_p}$, $\mathbf{D_b}$ and $\mathbf{D_s}$.

Method	F1-$\mathbf{D_{ext}}$	F1-$\mathbf{D_p}$	F1-$\mathbf{D_b}$	F1-$\mathbf{D_s}$
SC_SOBS A	0.026	0.009	0.093	0.028
SC_SOBS M	0.023	0.006	0.065	0.024
SC_SOBS LFA	0.142	0.211	0.097	0.143
SC_SOBS LFM	0.048	0.097	0.144	0.044
SENSE A	0.051	0.021	0.434	0.054
SENSE M	0.026	0.007	0.117	0.027
SENSE LFA	0.219	0.116	0.148	0.231
SENSE LFM	0.047	0.042	0.315	0.049
FGCD-T	0.313	0.142	0.14	0.333
FGCD-S	0.405	0.002	0.181	0.432
FCDN	0.571	0.35	**0.629**	0.6
DFCD	0.573	0.361	0.628	0.603
DFCD+FT	0.607	0.365	0.603	0.634
DFCD+FT+BS	**0.616**	**0.376**	0.586	**0.641**

nificantly improve the performance of the state-of-the-art FGCD-S [12] by 52.10% in terms of F1-measure (an absolute increase of 0.211). Since the images of $\mathbf{D_p}$ dataset suffer from uncontrollable illuminations and camera misalignment in outdoor scenes, all compared methods report poor results in the third column of Table 2. Compared with FGCD-T [12], DFCD+FT+BS obtains significantly improvement about 164.8% relative F1-measure improvement (an absolute increase of 0.234). As shown in the fourth column of Table 2 for dataset $\mathbf{D_b}$, conventional change detection methods SC_SOBS [28] and SENSE [33] perform better on $\mathbf{D_b}$ dataset than on the other three datasets (SENSE A [33] achieves F1-measure of 0.434). However, fine-grained change detector FGCD-S [12] performs bad, only obtaining an F1-measure of 0.181. In contrast, our DFCD+FT+BS significantly improves the F1-measure over FGCD-S [12] by 223.76% (an absolute increase of 0.405) on $\mathbf{D_b}$. In the last column of Table 2, FGCD-S [12] shows the best performance on $\mathbf{D_s}$ than on other three datasets (with F1-measure of 0.432). However, DFCD+FT+BS obtains an improvement of 48.38% over FGCD-S [12], with an absolute increase of 0.209.

Fig. 5 plots the PR curves for comparative methods. Note that not all the methods can have PR curves plotted, for instance SC_SOBS variants and DFCD+FT+BS, which generate hard segmentation results. As shown in Fig. 5, our method outperforms the state-of-the-art change detection and fine-grained change detection methods. We also note that in Fig. 5 (b) all curves have low values. This tells us that the outdoor scenes are very challenging for fine-grained change detection. Fig. 6 shows fine-grained change detection examples for three kinds of real-world scenes. Our detection results shows better precision than others.

It is clear that fine-tuning is helpful to improve the F1-measure, and confidence boosting can further improve the F1-measure. We also calculate the average F1-measure across the four test datasets for our methods and FGCD [12]. Our DFCD achieves an average score of 0.569; DFCD+FT achieves 0.595; and DFCD+FT+BS achieves 0.601. On the other hand, FGCD-S [12] reports 0.359, while FGCD-T [12] is 0.285. In comparison, DFCD+FT+BS obtains an improvement of 67.41% over FGCD-S [12], with an absolute increase of 0.242.

4.3. Discussion

4.3.1 Importance of change augmentation

To verify the importance of the scene-aware minute change augmentation (Sec.3.1), we train a FCDN with 12,349 training samples without data augmentation. From Table 3, we can clearly see that scene-aware data augmentation significantly improves the performance. Note, such strategy is generally helpful for other CNN-based training tasks.

Table 3. Comparison of F1-measure of FCDN with (W) or without (WO) scene-aware minute change augmentation. $\mathrm{FGCD}_{\mathrm{ext-Sp}}$ denotes training sample without data augmentation.

-	$\mathbf{D_p}$	$\mathbf{D_b}$	$\mathbf{D_s}$	$\mathbf{D_{ext}}$
FCDN WO	0.007	0.163	0.425	0.373
FCDN W	0.35	0.629	0.6	0.571
$\mathrm{FGCD}_{\mathrm{ext-Sp}}$	100	479	295	11475

4.3.2 Parameters robustness

From the experiments, we find that FGCD [12] is sensitive to parameters. As shown in Fig. 1, its detection results as well as F1-measure can change greatly. On the contrast, our method uses fixed parameters in coarse alignment on all datasets and achieves the best detection performance. In other words, our method is rather robust.

4.3.3 Timing Performance

We now report the timing performance of our method and FGCD. The two methods are conducted on the same machine without using GPU acceleration. The size of test images is 457×643. As listed in Table 4, the coarse alignment just takes about 97s, almost half of the timing of FGCD. Our pose correction spends 8.4 more seconds, which is the computation time of PCN. For the change detection step, FGCD-T costs 288.7s while our DFCD uses 3.6s, which is about 80 times faster. FGCD-S and DFCD+FT+BS are a bit slower than FGCD-T and DFCD, respectively. Since our DFCD+FT has the same test process with DFCD, we only show the timing for DFCD. On the whole test dataset, the running time are $3,805.7$s for DFCD and $26,213.9$s for FGCD. Our method is 6.9 times faster than FGCD.

5. Conclusion

In this paper, we have proposed DeepFCD, a reliable fine-grained change detection model under variant imaging

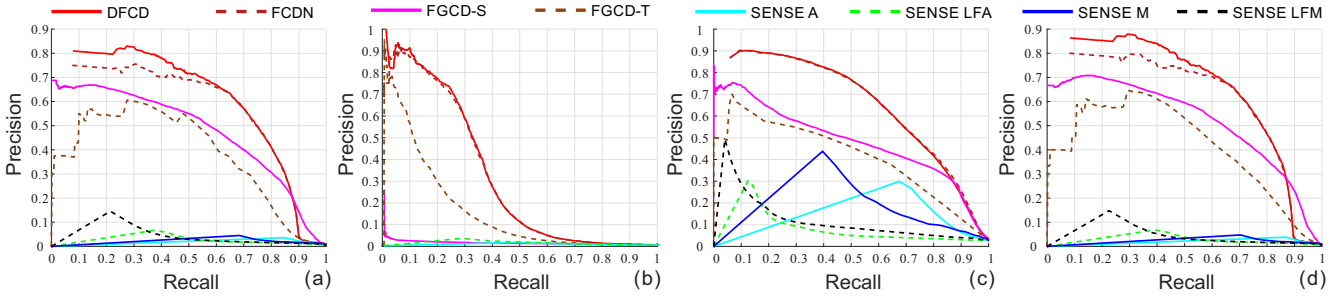

Figure 5. Comparisons of PR curves on $\mathbf{D_{ext}}$ (a), $\mathbf{D_p}$ (b), $\mathbf{D_s}$ (c) and $\mathbf{D_b}$ (d), respectively.

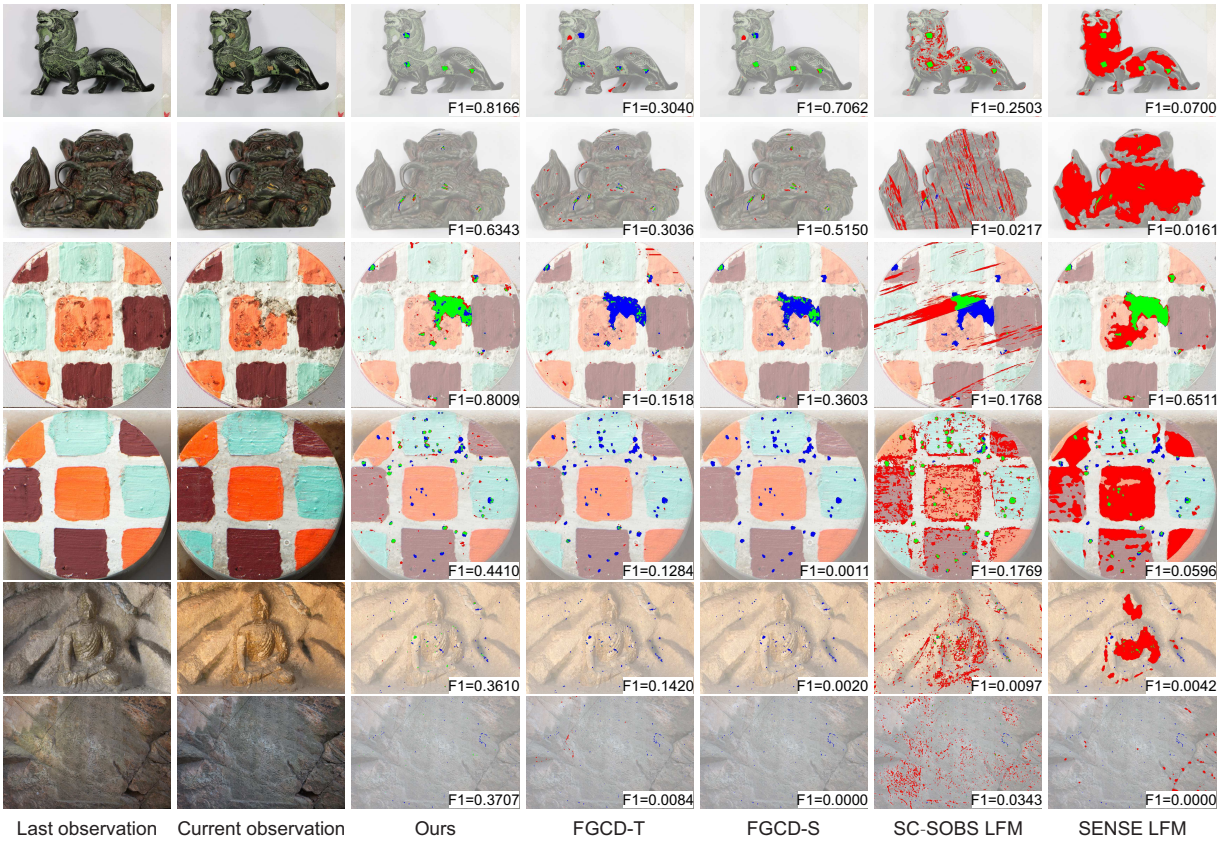

Figure 6. Comparisons of different fine-grained change detectors on several real-world scenes. Green color represents true positive, red color denotes false positive, blue color denotes false negative.

Table 4. Time comparison of our method and FGCD [12].

Method	Coarse Alignment	Pose Correction	Change Detection
FGCD-T	188.8s	55.1s	288.7s
FGCD-S	188.8s	55.1s	289.4s
DFCD	97.1s	63.5s	3.6s
DFCD+FT+BS	97.1s	63.5s	4.7s

conditions. To our best knowledge, this is the first practical minute change detector using deep learning scheme and exhibiting satisfactory generalization power, superior accuracy and reliability on challenging real-world data. Our major contributions are three-fold. First, we propose a hybrid deep network model, i.e., DFCD, taking coarsely aligned images as input and outputting fine-grained change map directly, which realizes end-to-end minute change detection via DNN and significantly improves the robustness to parameters and generalization power of state-of-the-art methods. Second, we help to construct a larger benchmark dataset of real-world scenes with both artificial and natural fine-grained changes. We also present feasible scene-aware minute change augmentation to further guarantee the amount of effective training samples. Third, we present a spatial alignment layer and corresponding error backpropagation method to enable the joint fine-tuning of pose correction and fine-grained change detection networks on valuable real data. In near future, we plan to explore more effective training strategies and to realize detection confidence boosting and accurate lighting correction via proper deep network models.

References

[1] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31, 2011.

[2] J. T. Barron. Convolutional color constancy. In *ICCV*, 2015.

[3] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE TPAMI*, 37(8):1670–1687, 2015.

[4] S. Bianco, C. Cusano, and R. Schettini. Color constancy using cnns. In *CVPRW*, 2015.

[5] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.

[6] H. Dai, W. Feng, L. Wan, and X. Nie. L0 co-intrinsic images decomposition. In *ICME*, 2014.

[7] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. *NIPS*, 2016.

[8] A. Dosovitskiy, P. Fischery, E. Ilg, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox, et al. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015.

[9] I. Eden and D. B. Cooper. Using 3d line segments for robust and efficient change detection from multiple noisy images. In *ECCV*, 2008.

[10] C.-Y. Fang, S.-W. Chen, and C.-S. Fuh. Automatic change detection of driving environments in a vision-based driver assistance system. *IEEE TNN*, 14(3):646–657, 2003.

[11] W. Feng, F. P. Tian, Q. Zhang, and J. Sun. 6d dynamic camera relocalization from single reference image. In *CVPR*, 2016.

[12] W. Feng, F. P. Tian, Q. Zhang, N. Zhang, L. Wan, and J. Sun. Fine-grained change detection of misaligned scenes with varied illuminations. In *ICCV*, 2015.

[13] D. Fourure, R. Emonet, E. Fromont, D. Muselet, A. Trémeau, and C. Wolf. Mixed pooling neural networks for color constancy. In *ICIP*, 2016.

[14] S.-B. Gao, K.-F. Yang, C.-Y. Li, and Y.-J. Li. Color constancy using double-opponency. *IEEE TPAMI*, 37(10):1973–1985, 2015.

[15] A. Gijsenij, T. Gevers, and J. Van De Weijer. Computational color constancy: Survey and experiments. *IEEE TIP*, 20(9):2475–2489, 2011.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[17] L. Giustarini, R. Hostache, P. Matgen, G. J.-P. Schumann, P. D. Bates, and D. C. Mason. A change detection approach to flood mapping in urban areas using terrasar-x. *IEEE T-GRS*, 51(4):2417–2430, 2013.

[18] L. Giustarini, R. Hostache, P. Matgen, J. P. Schumann, P. D. Bates, and D. C. Mason. A change detection approach to flood mapping in urban areas using terrasar-x. *IEEE TGRS*, 51(4):2417–2430, 2013.

[19] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. changedetection.net: A new change detection benchmark dataset. In *CVPRW*, 2012.

[20] L. Grady. Random walks for image segmentation. *IEEE TPAMI*, 28(11):1768–1783, 2006.

[21] D. Hauagge, S. Wehrwein, K. Bala, and N. Snavely. Photometric ambient occlusion. In *CVPR*, 2013.

[22] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CVPR*, 2016.

[23] S. H. Khan, X. He, M. Bennamoun, F. Porikli, F. Sohel, and R. Togneri. Weakly supervised change detection in a pair of images. *arXiv*, 2016.

[24] J. Kim, J. K. Lee, and K. M. Lee. Deeply-recursive convolutional network for image super-resolution. *CVPR*, 2016.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[26] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE TPAMI*, 33(5):978–994, 2011.

[27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[28] L. Maddalena and A. Petrosino. The SOBS algorithm: what are the limits? In *CVPRW*, 2012.

[29] X. Nie, W. Feng, L. Wan, H. Dai, and C.-M. Pun. Intrinsic image decomposition by hierarchical L0 sparsity. In *ICME*, 2014.

[30] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[31] K. Sakurada, T. Okatani, and K. Deguchi. Detecting changes in 3D structure of a scene from multi-view images captured by a vehicle-mounted camera. In *CVPR*, 2013.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2016.

[33] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin. SuBSENSE: A universal change detection method with local adaptive sensitivity. *IEEE TIP*, 24(1):359–373, 2015.

[34] S. Stent, R. Gherardi, B. Stenger, and R. Cipolla. Precise deterministic change detection for smooth surfaces. In *WACV*, 2016.

[35] D. Sun, S. Roth, and M. J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *IJCV*, 106(2):115–137, 2014.

[36] A. Taneja, L. Ballan, and M. Pollefeys. Image based detection of geometric changes in urban environments. In *ICCV*, 2011.

[37] A. Taneja, L. Ballan, and M. Pollefeys. City-scale change detection in cadastral 3d models using images. In *CVPR*, 2013.

[38] T. Tasdizen, E. Jurrus, and R. T. Whitaker. Non-uniform illumination correction in transmission electron microscopy. In *MICCAIW*, 2008.

[39] A. Vedaldi and K. Lenc. MatConvNet – convolutional neural networks for Matlab. In *ACMM*, 2015.

[40] K.-F. Yang, S.-B. Gao, and Y.-J. Li. Efficient illuminant estimation for color constancy using grey pixels. In *CVPR*, 2015.

[41] O. Yousif and Y. Ban. Improving urban change detection from multitemporal sar images using PCA-NLM. *IEEE T-GRS*, 51(4):2032–2041, 2013.