Did it change?
Learning to Detect Point-of-Interest Changes for Proactive Map Updates

Jérôme Revaud[†] Minhyeok Heo[§] Rafael S. Rezende[†] Chanmi You[§] Seong-Gyun Jeong[§]

[†]NAVER LABS Europe [§]NAVER LABS

Abstract

Maps are an increasingly important tool in our daily lives, yet their rich semantic content still largely depends on manual input. Motivated by the broad availability of geo-tagged street-view images, we propose a new task aiming to make the map update process more proactive. We focus on automatically detecting changes of Points of Interest (POIs), specifically stores or shops of any kind, based on visual input. Faced with the lack of an appropriate benchmark, we build and release a large dataset, captured in two large shopping centers, that comprises 33K geo-localized images and 578 POIs. We then design a generic approach that compares two image sets captured in the same venue at different times and outputs POI changes as a ranked list of map locations. In contrast to logo or franchise recognition approaches, our system does not depend on an external franchise database. It is instead inspired by recent deep metric learning approaches that learn a similarity function fit to the task at hand. We compare various loss functions to learn a metric aligned with the POI change detection goal, and report promising results.

1. Introduction

Maps have become a useful companion in our daily lives, as they provide a convenient and searchable representation of our physical world. To achieve map usability, raw geographical data is manually enriched with associated semantic information [18, 38]. This way, maps can be queried and explored in an effective and user-friendly manner.

Points of interest (POIs), *i.e.* well-localized landmarks that one may find useful or interesting, typically constitute an important part of this semantic content. POIs can be shops or stores of all kinds, including restaurants, cafés, banks, and so forth. Currently, most of this content is gathered and entered mostly by hand in the map database [18, 27, 41], but this process is tedious and expensive. This is problematic as map semantic content is alive by definition and, thus, subject to frequent changes. In addition, out-of-date POI information can be a source of accidents and user frustration [8, 37].

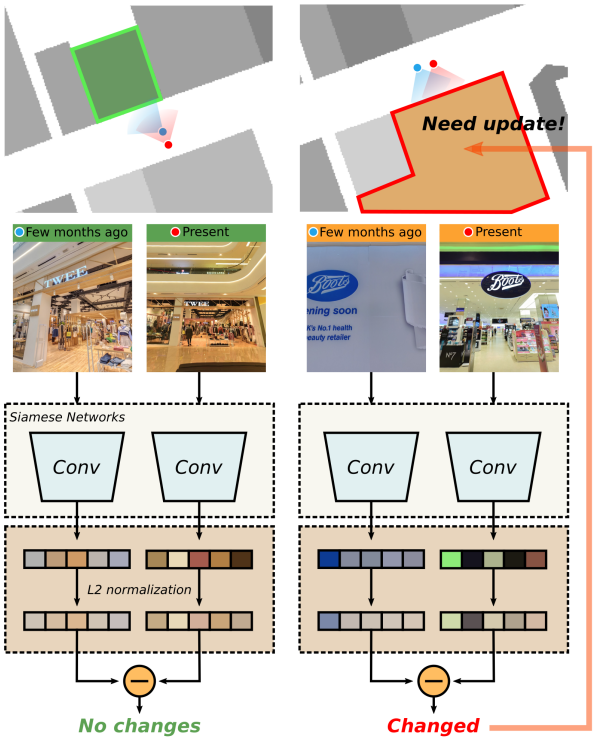


Figure 1. **Overview of the proposed system.** We detect changes of Points-of-Interest (POIs) as a first step towards the long-term goal of fully-automated map updates. To that purpose, we compare two sets of images captured at different moments. Image pairs corresponding to the same physical location are first formed and then compared using specially trained embeddings extracted by a deep network. The comparison is simply carried out as an inner-product between the two ℓ_2-normalized embeddings.

In this paper, we ask the following question: Could the map maintenance burden be alleviated by leveraging recent advances in computer vision? We notice that, much like humans, who unconsciously build a mental map of their surrounding relying heavily on their ocular sense, machines should also be able to perform semantic mapping-related

(a) POI appearance (b) POI replacement (c) POI disappearance (d) no change

Figure 2. **The different cases of POI changes.** The first three columns show the cases we wish to detect. The last case is not a POI change, but is challenging due to the lack of clearly identifiable brand or logo and the strong viewpoint and lighting changes.

tasks from visual input. We make one step towards the ambitious goal of fully-automated map maintenance and tackle the detection of POI changes based on spatio-temporally localized scene photographs. This task more specifically consists in notifying a database operator whenever a POI is changed, which happens when a new POI appears, a POI is replaced by another one or a POI disappears, see Fig. 2.a–c. It implies comparing two sets of images captured at distinct moments and is challenging for several reasons. First, the comparison must be robust to various sources of noise, *e.g*. lighting, reflection, shadows, occlusions and viewpoint changes. Image poses and scaling can dramatically differ between the two captures, as exemplified in Fig. 2.d. Second, substantial intrinsic variations of POI appearance happen over time due to seasonal changes, special sales, etc. Third, the system is expected to produce an output aligned with the final goal, *i.e*. a list of geographical POI spots likely to have changed, which requires a novel framework.

This paper is also motivated by the recent explosion of initiatives to capture photos spanning all areas of the real-world: nowadays, online map services offer immersive experiences regarding street views [3]. Given the existence and availability of this data [4, 48], it is somewhat surprising that no work, to the best of our knowledge, has yet focused on this important aspect of map content update. One potential reason for this absence is the lack of an appropriate benchmark. Although new datasets related to urban localization [4, 5, 48], place attributes [54–56] and landmarks [7, 21, 29–31] have recently emerged, information related to both POIs and time-stamps is unavailable.

As a first contribution, we provide a new and challenging annotated dataset to bring this interesting task to the attention of researchers. The dataset is composed of thousands of photographs, captured in two large shopping malls, each comprising hundreds of POIs. By focusing on indoor images, which are simpler to analyze and parse than outdoor scenes, we isolate the problem at hand from other out-of-scope challenges. Since change detection assumes temporality, photographs are divided into two groups captured at different time-stamps separated by several months. We also provide the precise geo-localization of each photo, estimated at acquisition time using a LIDAR. Overall, our dataset comprises 578 POI instances and more than 33K images with relative 6-DoF camera pose information. More importantly, we have annotated POI changes at two different levels: at the image-level, but also at a geographical level.

Our second contribution is a novel and generic approach for POI change detection. Our system is based on three key stages: (1) temporally distant images are matched to form pairs if their poses overlap; (2) for each pair, the images are compared to detect POI changes; (3) pairwise predictions are aggregated at the scale of the entire venue. The proposed system only makes loose assumptions on the quality, content and exhaustiveness of the input photographs. Remarkably, it is able to detect changes even in the absence of clearly identifiable logos or signage.

Our third contribution is a benchmark of several alternative techniques for the image pair comparison step (second stage above). These range from keypoint-matching with geometric verification to state-of-the-art deep learning techniques for metric learning, see Fig. 1. The method that works best, based on deep embedding trained with a triplet loss, achieves promising results.

The rest of the paper is organized as follows. We discuss related work in Section 2 and present our new dataset, coined *MallScape*, in Section 3. We describe the proposed overall approach for POI change detection in Section 4 and benchmarks different methods and options in Section 5.

2. Related Work

Change detection in images has been a long topic of interest [35] in multiple domains such as medical imaging [28], remote sensing [11, 12], camera surveillance [15, 23, 53] or aerial image analysis [2, 9, 22, 42]. In the medical field, change detection has mainly been seen as a way of monitoring the appearance of tumors or other anomalies in X-ray or MRI images. To that aim, several images taken at different moments are first carefully aligned and then pixel-wise compared [28]. Similar problems arise in the context of aerial images [2, 9, 42] and remote sensing [11, 12], where the goal is again to observe the evolution of specific regions or constructions between images precisely aligned beforehand. These approaches rely on specific constraints and conditions and are not easily generalizable to other cases [15].

Closer to our problem, the task of detecting structural changes in outdoor urban scenes has recently developped [2, 9, 16, 22, 42]. Similar to our case, these methods feed on geo-localized photographs, typically captured by vehicle-mounted cameras equipped with GPS tracking devices. By comparing pairs of pictures of the same location captured at different moments, they aim to predict a binary pixel mask indicating structural changes, *e.g.* displaced objects or road works. For successful predictions, image pairs must first undergo an accurate aligment procedure using complex and error-prone 3D reconstruction techniques [2, 6]. In this work, we do not make assumptions on the precise alignment of images. More importantly, these methods are blind to the nature of the change and it is unclear if they would detect a POI change, which does not necessarily involves any structural changes (*e.g.* see Fig. 2.b). On the contrary they would incorrectly detect structural changes due to periodical store-front rearrangements or special events like Christmas. Finally, these methods are optimizing pixel-level metrics that are not suited to the goal of map updates. We detect and aggregate POI changes at a geographical scale inside entire venues.

Logo and franchise detection. Another category of work has focused on recognizing POIs, or rather what makes them identifiable: logos or representative symbols of their brands. Deep approaches for logo detection [44, 45] have recently outperformed former hand-crafted approaches [39, 40] and larger datasets have been collected by fetching images on social media such as Flickr and Twitter [49]. While it is theoretically possible to detect POI changes based on tracking brand logos or signage over time, it is subject to two major issues: (i) it implies that an exhaustive database of logos and brands is available and up-to-date, which is unrealistic given that new brands appear every day, and (ii) it overlooks the fact that in practice, due to occlusions, changes in viewpoint and image framing, logos and signages are often absent (*e.g.* see Fig. 2.d). Moreover many POIs do not belong to any franchise. In this work, we propose a framework that makes none of the two above assumptions and is still able to accurately detect POI changes.

Image retrieval and metric learning. Our approach is inspired by recent progress in image retrieval. In particular, recognizing places despite appearance and illumination changes over time and seasons is typically cast as an image retrieval problem [4, 5, 48]. Image retrieval aims to define a distance measure between images so that, given an image query, similar images can be retrieved from a large collection [21, 29, 31]. Ideally, this distance metric should be invariant to semantically meaningless variations induced by lighting or viewpoint changes. Recent works have shown that this metric can be *learned* with supervision using deep Siamese networks [25]. In fact, deep metric learning has proven overwhelmingly effective for image retrieval [14, 34], person re-identification [51], fine-grained image classification [52], 3D object retrieval [20] and place recognition [5]. We follow this example and learn, using minimal supervision, a distance metric that fits POI change detection.

In more detail, deep metric learning consists of learning an embedding function that project images in a space where Euclidean distance is an acurate measure of their semantic similarity. Many variants of the objective loss function have been developed, *e.g.* the contrastive [17], double-margin [24], triplet [43], and quadruplet [10] losses. Each has its own specificity (see a review in [25, 50]) but they have the common goal of pushing apart points belonging to different classes in the embedding space while at the same time attracting points having the same label. We experiment with several of these techniques for our problem.

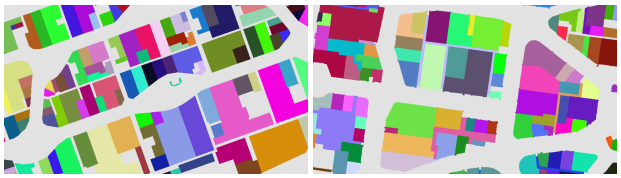
3. The MallScape dataset

In this section, we describe a new dataset specifically tailored to the task of POI change detection. It is composed of indoor scene photographs captured in real shopping centers. Each image comes with a precise 6 degrees-of-freedom (6-DoF) localization pose obtained by LIDAR. To observe real exemplars of POI changes, two distinct acquisition sessions separated by several months have been conducted. The dataset can be downloaded at [1].

Acquisition scenario. Our mapping device sweeps the venue and captures photos densely enough so that every portion of the wall appears at least in one photo in close-up, without any other special contraints. In order to ensure that POIs are well visible even when the device is very close to them, we mount cameras such that they pitch slightly upwards. This lazy acquisition scenario makes the capture system easier to implement and more scalable in realistic

Table 1. Summary of *MallScape-A* and *MallScape-B* datasets.

Dataset	*MallScape-A*					*MallScape-B*	**Total**
Floor	B1	1F	2F	3F	4F	B1	-
# of images	696	5640	5736	3912	156	17531	**33671**
# of POI	13	106	87	73	2	297	**578**
# of changes	0	6	6	4	0	6	**22**



(a) *MallScape-A* (b) *MallScape-B*

Figure 3. Parts of the floor maps corresponding to the first floor of *MallScape-A* (a) and the retail areas of *MallScape-B* (b). Each POI is represented by a distinct color.

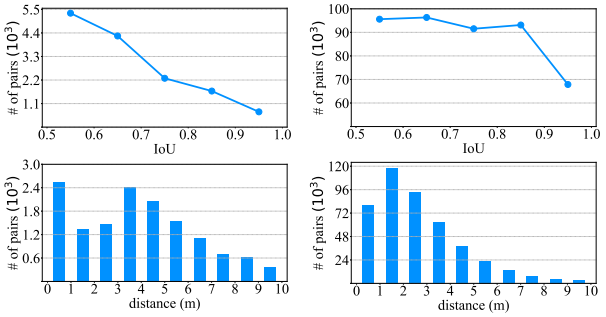
conditions. We now describe the two large shopping centers that have served for the acquisition campaign.

MallScape-A is a five-story building comprising 281 POIs, where the total retail floor area is about $460,000\ m^2$. We captured data twice with a 6-month time gap and took 360-degree panoramic images every 7 meters. Images were then mapped back to standard rectilinear camera lenses with 12 equidistant horizontal viewpoints in portrait mode, each having horizontal and vertical fields of view of 70 and 85 degrees, respectively.

MallScape-B covers about $144,000\ m^2$, and contains 297 POIs on a single underground floor. About 17K pictures were captured during two sessions separated by 3 months. Unlike the *MallScape-A* dataset, we used standard cameras equipped with fish-eye lenses, resulting in photographs in landscape mode with horizontal and vertical fields of view of respectively 107 and 70 degrees.

After the collection, images were carefully reviewed and annotated by semi-automatic methods in order to label the POI visible in each image as well as record POI changes. We provide with the dataset the labels of the POIs appearing in each image. Overall, the dataset contains a total of 578 POIs and 22 POI changes. Details about the number of images, number of POIs and number of POI changes are summarized in Table 1. Excerpts of the floor maps can be seen in Fig. 3, each color denoting a different POI.

Some image pairs showing the same POI at two different moments are exemplified in Fig. 5. Statistics computed over the camera pose for all such pairs are displayed in Fig. 4. On *MallScape-A*, the average distance between the two cameras is 4 meters, while the average intersection-over-union of the visual content is 0.66 (computation of these metrics is detailed later in Section 4.2). This shows that image pairs have substantially different viewpoints overall.



(a) *MallScape-A* (b) *MallScape-B*

Figure 4. Statistics on positive image pairs for both shopping centers *MallScape-A*and *MallScape-B*. Positive pairs, formally defined in Eq. (7), are the ones showing the same part of the same POI. We present histograms of the geometric overlap (computed according to Eq. (5)) between these pairs and their geometric distances (in meters). Cameras can be up to 10 meters away, resulting in significant viewpoint difference.

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

Figure 5. Examples of matching images pairs (each column shows the same place seen from different viewpoints and moments). Our proposed deep metric learning-based approach can accomodate images showing a single POI (a), multiple POIs (b), and no POI at all (c). We show in column (d) an example of dramatic appearance variations due to advertisement.

4. POI Change Detection

We describe in this section the problem that we want to solve. We then propose a generic metric learning approach to solve it, and discuss different options for training the central pairwise image similarity.

4.1. Problem formulation

We are interested in automatically determining, for each location within a certain area, if a POI at this location has changed or not over a period of time. Let D^t denote a dataset of geo-localized images captured at time t[1], *i.e.*

[1] for simplicity, we assume simultaneity of the capture.

$D^t = \{(I_i^t, \Theta_i^t)\}_i$ where I_i^t is an image and Θ_i^t its associated 6-DoF camera pose. We further assume the existence of a second dataset $D^{t'}$ captured at a different time $t' > t$. Note that we do not make any assumption on the correspondences between images and poses between D^t and $D^{t'}$ except that both image sets are captured in the same area.

Our first goal is to learn a function ς that predicts the similarity between two localized images:

$$\varsigma : (I_i^t, \Theta_i^t) \times (I_j^{t'}, \Theta_j^{t'}) \mapsto [0, 1]. \quad (1)$$

We design ς such that the similarity is high when the two images show the same POI, and low otherwise. We investigate various ways of accomplishing this goal in the next sections.

Eventually, we want to find all modifications or alterations of POIs in the target area, regardless of the reason. Formally, our final goal is to score each potential POI location with a corresponding change likelihood. Let $g : \mathcal{P} \mapsto [0, 1]$ denote such POI change scoring function, where $\mathcal{P} \subset \mathbb{R}^3$ denotes the coordinate space of all locations addressed by latitude, longitude and elevation. In practice, we implement $g(\cdot)$ straightforwardly by max-pooling the pairwise similarity scores output by $\varsigma(\cdot)$:

$$g(p) = 1 - \max_{\substack{\Theta_i^t \in \mathcal{V}^t(p) \\ \Theta_j^{t'} \in \mathcal{V}^{t'}(p)}} \varsigma\left((I_i^t, \Theta_i^t), (I_j^{t'}, \Theta_j^{t'})\right), \quad (2)$$

where $\mathcal{V}^t(p)$ is the visibility set of p, *i.e.* the set of image poses $\{\Theta_i^t\}_i$ from which one can directly see location p (and likewise for $\mathcal{V}^{t'}$). Otherwise stated, function $g(p)$ predicts a POI change if not a single pair of images showing location p has high similarity.

The final decision of whether or not a given map spot p has undergone a change of POI is made by comparing $g(p)$ with a threshold τ. In practice, we beforehand smooth $g(\cdot)$ using a Gaussian kernel of spatial radius $\sigma = 2$ meters to remove noise. In our experiments, we evaluate the accuracy of $g(\cdot)$ after thresholding as well as the average precision of the spots ranked by $g(\cdot)$ (see Section 5.2).

4.2. Visibilty sets and pose-based constraints

The formulation above assumes that we can compute the set of locations visible for each image pose. We achieve this by leveraging the floor map $\mathcal{M}$ that specifies the positions of all walls. Specifically, we use a ray-casting technique to compute the set of wall points that are visible from each camera pose. The process is illustrated in Fig. 6. Each ray starts from the camera center at $P(\Theta) \in \mathbb{R}^3$, passes through the camera lens and continues until it hits a wall. The set of rays form a conical region of width corresponding to the field of view of the camera. Each hit point is labeled as

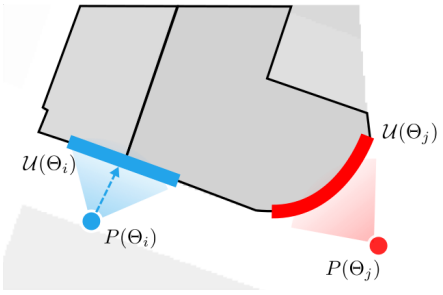


Figure 6. The knowledge of the floor map combined with simple ray-casting technique allows us to compute the wall points $\mathcal{U}(\Theta)$ visible from camera pose Θ.

a potential POI location p (or more accurately, a potential store-front facade), and their union constitutes $\mathcal{U}(\Theta)$. The unions of the visible facades from all viewpoints defines the set of all potential POI locations:

$$\mathcal{P} = \bigcup_i \mathcal{U}(\Theta_i). \quad (3)$$

Conversely, the set of all camera poses seeing point p is stored as its visibility set $\mathcal{V}(p) = \{\Theta_i | p \in \mathcal{U}(\Theta_i)\}$.

To simplify the similarity function ς (Eq. (1)), we beforehand exclude all image pairs having inconsistent poses:

$$\varsigma((I_i^t, \Theta_i^t), (I_j^{t'}, \Theta_j^{t'})) = \begin{cases} s(I_i^t, I_j^{t'}) & \text{if is\_valid}(\Theta_i, \Theta_j), \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

where s is a trained similarity function purely relying on visual input and defined in Section 4.3. An image pair is deemed consistent, or valid, if its overlap is substantial and if both images are close enough to a wall, as images far from store-fronts are generally poorly informative. First, the overlap between two images is computed as the intersection-over-union of their respective visibility sets:

$$O(\Theta_i, \Theta_j) = \frac{|\mathcal{U}(\Theta_i) \cap \mathcal{U}(\Theta_j)|}{|\mathcal{U}(\Theta_i) \cup \mathcal{U}(\Theta_j)|}, \quad (5)$$

Then, the average wall distance for a given image is computed based on $\mathcal{U}(\Theta)$ as $D(\Theta) = \frac{1}{|\mathcal{U}(\Theta)|} \sum_{p \in \mathcal{U}(\Theta)} \|p - P(\Theta)\|$ (in practice, we use 10 meters as distance threshold). These two conditions practically remove many irrelevant pairs in Eq. (2) and considerably speed up the inference.

4.3. Learning a similarity function

The goal of metric learning is to learn a similarity measure between images under some supervision. It has been applied successfully to various fields, such as image retrieval [14,32] and person re-identification [10,51]. In practice, it is often formulated as the task of learning an image

embedding function $f(I) = x \in \mathcal{X}$, with $\mathcal{X} \subset \mathbb{R}^N$ an ℓ_2-normalized embedding space of dimension N. Similarity s from Eq. (4) is computed as the inner product between two embeddings, *i.e.* we have

$$s(I_i, I_j) = \max\left(0, f(I_i)^\top f(I_j)\right). \tag{6}$$

We can now learn an embedding function that suits our definition of similarity. More specifically, let $y_i = \{o_1 \dots o_{m_i}\}$ and $y_j = \{o_1 \dots o_{m_j}\}$ denote the set of POIs visible in image I_i and I_j. We define the ground-truth similarity $Y(i, j)$ as follows:

$$Y(i,j) = \begin{cases} 1 & \text{if } O(\Theta_i, \Theta_j) > 0.4 \text{ and } |y_i \cap y_j| > 0, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

i.e. two images are not similar unless they show at least one common part of the same POI. Fig. 5 shows several examples of matching (*i.e.* positive) image pairs.

Loss function. We use a deep network to learn function $f(\cdot)$, trained with stochastic gradient descent. Gradients are computed at each iteration according to the agreement between the ground-truth Y and the current similarity, computed by Eq. (6), thanks to an appropriate loss function.

For instance, the contrastive loss [17, 33] denoted by L_c attracts positive pairs ($Y(i, j) = 1$) to each other while pushing negative pairs ($Y(i, j) = 0$) apart:

$$\begin{aligned} L_c(i,j) = {} & Y(i,j)\,(1 - s(I_i, I_j)) \\ & + (1 - Y(i,j)) \max\left(0,\ s(I_i, I_j) - \tau_c\right), \end{aligned} \tag{8}$$

where τ_c is a similarity threshold between negative pairs below which the loss has no effect.

Ideally, we want to discriminate between changes and non-changes by thresholding $g(\cdot)$ with τ. The most direct way to achieve that is by penalizing any training image pairs that deviates from this behavior. This corresponds to the double-margin pairwise loss L_{dm} proposed in [24]:

$$\begin{aligned} L_{dm}(i,j) = {} & Y(i,j) \max\left(0,\ \left(\tau + \frac{m}{2}\right) - s(I_i, I_j)\right) \\ & + (1 - Y(i,j)) \max\left(0,\ s(I_i, I_j) - \left(\tau - \frac{m}{2}\right)\right). \end{aligned} \tag{9}$$

where m is a tunable margin. Alternatively, we can use the contrastive loss L_c with $\tau_c = \tau - \frac{m}{2}$.

Finally, another popular loss is based on image triplets [43]:

$$\begin{aligned} & L_t(i,j,k) = \max\left(0,\ m - s(I_i, I_j) + s(I_i, I_k)\right), \\ & \text{with } Y(i,j) = 1 \text{ and } Y(i,k) = 0. \end{aligned} \tag{10}$$

This loss has been shown to be easier to train because it only enforces the relative ordering of the positive (i, j) and negative (i, k) similarities [25, 50], whereas previous formulations enforce an absolute similarity threshold. While there is no guarrantee that the triplet loss will preserve a global threshold τ suitable for $g(\cdot)$, in practice we observe good performance.

The procedure for training is to repeatedly sample random pairs or triplets of images (depending on the loss), and to compute the loss for each. If the loss is non-zero, then the loss gradient is computed and serves to update the network weights. In practice, we sample as many positive pairs as negative ones to balance the training.

5. Experimental results

5.1. Protocol and metrics

We now introduce evaluation metrics and experimental protocols tailored to our problem and data. We perform the evaluation at two different levels: on image pairs (intermediate goal in Section 4.1) and at the geographical level (our final goal).

Ground-truth. The quality of the similarity measure on image pairs is evaluated with respect to the ground-truth $Y(i, j) \in \{0, 1\}$ defined in Eq. (7). For the geographical level, we annotate the set of potential POI locations $\mathcal{P}$ (Eq. (3)) as follows: each location $p \in \mathcal{P}$ belonging to a true POI change is marked as positive, and negative otherwise. For convenience we abuse notations and denote it as $Y(p) \in \{0, 1\}$ in the following. We now introduce the different metrics.

ROC curve. Detecting POI changes can be seen as a binary classification task: change ($Y(p) = 1$) vs. no change ($Y(p) = 0$). The ROC curve allows to measure the overall performance of a binary classifier at multiple thresholds. It is produced by computing the true and false positive rates (TPR and FPR, respectively) for all thresholds:

$$\mathrm{TPR}(\tau) = \frac{\sum_p \mathbb{I}[g(p) \geq \tau]}{\sum_p Y(p)}, \quad \mathrm{FPR}(\tau) = \frac{\sum_p \mathbb{I}[g(p) < \tau]}{\sum_p 1 - Y(p)}$$

We also report the area under the curve (AUC). We compute these metrics both for image pairs (denoted as pROC and pAUC) and for geographic locations (denoted as gROC and gAUC).

Average Precision (AP). Our final goal can also be formulated as globally ranking all map locations in terms of POI change likelihood. We measure the system performance in terms of ranking using the AP. The AP computed for image pairs is denoted as pAP and the geographic-level AP is denoted as gAP in the following.

Train and test splits. *MallScape* consists of two sub datasets captured in different venues. Despite the large number of dataset images, there are relatively few instances of POI changes. This can increase the variance of the noise in the evaluation metrics and negatively affect the evaluation. To smooth the performance, we therefore train and test on one and the other venue in turns and finally average the results:

- *Split 1*: train on *MallScape-A* and test on *MallScape-B*;
- *Split 2*: train on *MallScape-B* and test on *MallScape-A*

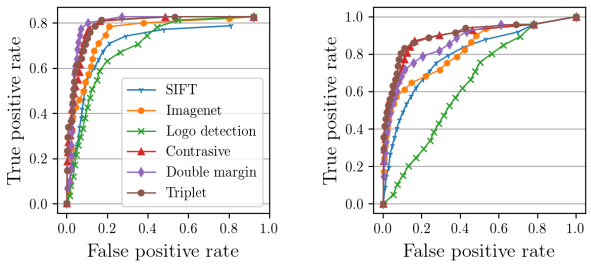
Since the styles of the two venues are significantly different, these splits also ensures that good performance is not due to training set overfitting.

5.2. Quantitative results

Implementation details All our models are formed by a ResNet-101 [19] backbone to which we append a global generalized-mean pooling layer [33]. The embedding dimension is $N = 2048$. After some preliminar experiments, we decide to use a weight decay of 0 and a learning rate of 10^{-5}, decreased by 2 every 2000 iterations. We employ standard data augmentation during training to increase generalization performance [19]. To improve training speed and test performance, we follow standard practice for pretraining and hard-negative mining [14, 32, 34]. We also tune the margin m and threshold τ_c parameters separately for each loss.

Results. We study the performance of the different loss functions presented in Section 4.3. We also compare to three baseline approaches. The first one relies on keypoint-matching followed by geometric verification [26] to compute the similarity s from Eq. (4) using a sigmoid. The second one is based on a state-of-the-art logo detector [46] able to recognize 352 common logos in real-world images. In this case, images are thus compared in terms of detected brands. Lastly, we also include the performance of embeddings extracted from the last convolutional layer of a network pretrained on ImageNet [47] for reference.

Results for all methods with all metrics are presented in Table 2. It is clear that off-the-shelf features trained on ImageNet produce poor embeddings, highlighting the importance of learning the metric. Similarly, SIFT-based features are unable to analyze the complex semantic changes involved in the POI detection task. In contrast, the logo-based baseline specifically focuses on semantic aspects that are key to the task, yet, it performs more poorly than the ImageNet baseline. After inspection, we find that many POI images do not contain any logos for which the detector was trained. These 352 known logos form, after all, a rather small part of all logos potentially appearing in real POI images. We therefore believe that approaches explicitly recognizing brands are rather impractical, as noted in Section 2.



(a) *Split* 1 (b) *Split* 2

Figure 7. ROC curves for all methods on both splits at the geographic level (gROC).

Unsurprisingly, the three methods based on metric learning (ML) perform similarly in terms of average pAUC and gAUC. However the quality of the ranking of POI scores (gAP) is significantly better for the triplet loss L_t. This is in agreement with previous findings that the triplet loss better behaves during training in general than pairwise losses [14, 50]. Interestingly, we note that image-level metrics can be in complete disagreement with geographical metrics. This is for instance the case on the second split where ImageNet features yield, at the same time, the best pAP score and the worse gAP score by far among deep methods. This underlines the fact that detecting POI changes should be evaluated with respect to its final goal (at the geographical level) instead of relying on simpler image-level criteria (*e.g.* pixelwise or image-level metrics applied on image pairs).

We also plot the gROC curve, for both splits, in Fig. 7. It is interesting to note that the difficulty of the splits is not homogeneous, and that methods behave quite differently on both splits. While we expect the double-margin loss to perform better in term of gAUC, a detection metric, due to the perfect alignment with the task at hand (Section 4.3), this is only true for the first split. In contrast, the triplet and contrastive losses yield more stable performance on both splits.

5.3. Qualitative results on map database

We present quantitative results of POI change predictions generated at the map level for each of the learned models in Fig. 8. Each row shows a part of the floop map augmented with the geographical likelihood of POI changes, as output by $g(p)$ from Eq. (2). The likelihood is color-coded from green (no change) to red (change).

We also show some image pairs corresponding to the POIs pinned on the map. Some of them present dramatic viewpoint changes, e.g. Fig. 8.a. The learned approaches most correctly predict the absence of changes up to some extent, *e.g.* in the case of first row. Conversely, the viewpoint difference in the second row of Fig. 8.a exceeds the tolerance range of the models and results in a false prediction of POI change.

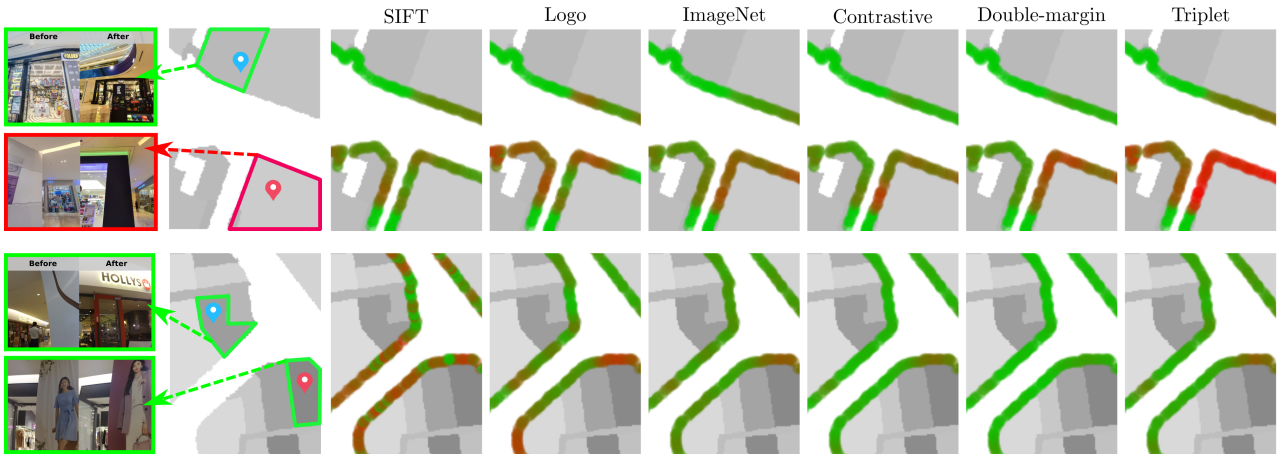


Figure 8. We present some examples of POI change likelihood output at the geographical level according to Eq. (2). The two rows are from *MallScape-A* (top) and *MallScape-B* (bottom), respectively. We show two corresponding image pairs from the locations pinned on the partial floor map (a). The ground-truth and the change likelihood are color-coded as *changed* and *not-changed*.

Table 2. Performance evaluation for all methods and all metrics (see text for details).

	Parameters		**Split 1**				**Split 2**				**Overall**			
	name	value	pAUC	pAP	gAUC	gAP	pAUC	pAP	gAUC	gAP	pAUC	pAP	gAUC	gAP
Local descriptor														
SIFT [26]+RANSAC	-	-	0.533	0.080	0.731	0.101	0.823	0.101	0.795	0.167	0.678	0.091	0.763	0.134
Logo detection														
YOLOv2 [36]+CAL [46]	-	-	0.690	0.005	0.711	0.077	0.642	0.003	0.638	0.058	0.666	0.004	0.675	0.068
Global representations														
ImageNet [13]	-	-	0.932	0.386	0.758	0.201	0.932	**0.245**	0.827	0.391	0.932	0.316	0.793	0.296
Deep metric learning														
ML+L_c (8)	τ_c	0.5	0.970	**0.591**	0.787	0.393	0.959	0.214	0.898	0.508	0.965	0.403	0.843	0.451
ML+L_{dm} (9)	m, τ	0.1, 0.1	0.961	0.556	**0.793**	0.330	0.930	0.105	0.868	0.408	0.946	0.331	0.831	0.369
ML+L_t (10)	m	0.1	**0.973**	0.582	0.786	**0.412**	**0.970**	0.228	**0.905**	**0.557**	**0.972**	**0.405**	**0.846**	**0.485**

Since our framework allows to output results directly at a geographical level, visualization is easy and straightforward. A human operator can rapidly understand the positions of all POI changes in a glance and update them accordingly, which can greatly ease the update process. Ultimately, the update process could become fully automatic if logo or franchise recognition [44, 45, 49] would be performed on the corresponding POI images.

6. Conclusion

We have presented a novel generic approach, based on the deep metric learning framework, that can detect POI changes from a set of spatiotemporally localized scene photographs. Several metric learning formulations were thoroughly evaluated and tested, which confirmed their overall effectiveness for this problem. In particular, the triplet loss seems best for this problem from an empirical perspective.

To enable training and evaluation, we have introduced a new dataset dedicated to the POI change detection task. It contains thousands of images and hundreds of POIs, making it suitable for training deep models in realistic settings. Not only can this dataset also serve as a benchmark suite for other researchers interested in this task, but we believe that it can also help to further develop new exciting tasks related to automatic map creation and maintenance, thanks to the rich information encompassed in the dataset.

We indeed acknowledge that the approach proposed here calls for further research. For instance, it does not allow us to understand if a single photo contains several POIs, and if so, what their boundaries are. Yet automatic shop segmentation is an important milestone to understand the spatial extent of each POI and thus to map them to a geographic location, which would certainly help to better detect and localize POI changes. We leave these issues for future work.

Acknowledgements This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.R0132-15-1005, Content visual browsing technology in the online and offline environments)

References

[1] http://rebrand.ly/mallscape. 3

[2] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi. Street-view change detection with deconvolutional networks. *Auton. Robots*, 42:1301–1322, 2016. 3

[3] D. Anguelov, C. Dulong, D. Filip, C. Frueh, S. Lafon, R. Lyon, A. Ogale, L. Vincent, and J. Weaver. Google street view: Capturing the world at street level. *Computer*, 43(6):32–38, June 2010. 2

[4] A. T. R. Arandjelovic, J. S. M. Okutomi, and T. Pajdla. 24/7 place recognition by view synthesis. In *IEEE CVPR*, 2015. 2, 3

[5] R. Arandjelovi, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE CVPR*, 2016. 2, 3

[6] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera. Are you able to perform a life-long visual topological localization? *Autonomous Robots*, 42(3):665–685, Mar 2018. 3

[7] Y. Avrithis, Y. Kalantidis, G. Tolias, and E. Spyrou. Retrieving landmark and non-landmark images from community photo collections. In *ACM Multimedia*, 2010. 2

[8] J. Baus, K. Cheverst, and C. Kray. *A Survey of Map-based Mobile Guides*, pages 193–209. Springer Berlin Heidelberg, 2005. 1

[9] K.-T. Chen, F.-E. Wang, J.-T. Lin, F.-H. Chan, and M. Sun. The world is changing: Finding changes on the street. In *ACCV Workshop*, 2016. 3

[10] W. Chen, X. Chen, J. Zhang, and K. Huang. Beyond triplet loss: A deep quadruplet network for person re-identification. In *IEEE CVPR*, 2017. 3, 5

[11] R. C. Daudt, B. L. Saux, and A. Boulch. Fully convolutional siamese networks for change detection. In *IEEE ICIP*, 2018. 3

[12] R. C. Daudt, B. L. Saux, A. Boulch, and Y. Gousseau. High resolution semantic change detection. *arXiv:1810.08452*, 2018. 3

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE CVPR*, 2009. 8

[14] A. Gordo, J. Almazán, J. Revaud, and D. Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016. 3, 5, 7

[15] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. Changedetection.net: A new change detection benchmark dataset. In *IEEE CVPR Workshop*, 2012. 3

[16] E. Guo, X. Fu, J. Zhu, M. Deng, Y. Liu, Q. Zhu, and H. Li. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. *arXiv:1810.09111*, 2018. 3

[17] R. Hadsell, S. Chopra, and Y. Lecun. Dimensionality reduction by learning an invariant mapping. In *IEEE CVPR*, 2006. 3, 6

[18] M. Haklay and P. Weber. Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008. 1

[19] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE CVPR*, 2016. 7

[20] X. He, Y. Zhou, Z. Zhou, S. Bai, and X. Bai. Triplet-center loss for multi-view 3d object retrieval. In *IEEE CVPR*, 2018. 3

[21] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008. 2, 3

[22] J. Košecka. Detecting changes in images of street scenes. In *ACCV*, 2012. 3

[23] L. A. Lim and H. Y. Keles. Learning multi-scale features for foreground segmentation. *arXiv:1808.01477*, 2018. 3

[24] J. Lin, O. Morere, A. Veillard, L. Duan, H. Goh, and V. Chandrasekhar. Deephash for image instance retrieval: Getting regularization, depth and fine-tuning right. In *ICMR*, 2017. 3, 6

[25] T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 2009. 3, 6

[26] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 7, 8

[27] L. N. Mummidi and J. Krumm. Discovering points of interest from users' map annotations. *GeoJournal*, 72(3):215–227, 2008. 1

[28] A. Naitsat, E. Saucan, and Y. Zeevi. A differential geometry approach for change detection in medical images. In *ISCBMS*, 2017. 3

[29] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. Largescale image retrieval with attentive deep local features. In *IEEE CVPR*, 2017. 2, 3

[30] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE CVPR*, 2007. 2

[31] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *IEEE CVPR*, 2008. 2, 3

[32] F. Radenović, G. Tolias, and O. Chum. Cnn image retreival learns from BoW: Unsupervised fine-tuning with hard examples. In *ECCV*, 2016. 5, 7

[33] F. Radenović, G. Tolias, and O. Chum. Fine-tuning CNN image retrieval with no human annotation. *TPAMI*, 2018. 6, 7

[34] F. Radenovi, A. Iscen, G. Tolias, Y. Avrithis, and O. Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *IEEE CVPR*, 2018. 3, 7

[35] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: A systematic survey. *TIP*, 2005. 3

[36] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In *IEEE CVPR*, 2017. 8

[37] K. Rehrl, E. Häusler, R. Steinmann, S. Leitinger, D. Bell, and M. Weber. *Pedestrian Navigation with Augmented Reality, Voice and Digital Map: Results from a Field Study assessing Performance and User Experience*, pages 3–20. Springer Berlin Heidelberg, 2012. 1

[38] D. Reilly, M. Rodgers, R. Argue, M. Nunes, and K. Inkpen. Marked-up maps: Combining paper maps and electronic information resources. *Personal Ubiquitous Comput.*, 10(4):215–226, 2006. 1

[39] J. Revaud, M. Douze, and C. Schmid. Correlation-Based Burstiness for Logo Retrieval. In *ACM Multimedia*, 2012. 3

[40] S. Romberg, L. G. Pueyo, R. Lienhart, and R. van Zwol. Scalable logo recognition in real-world images. In *ICMR*, 2011. 3

[41] M. Ruta, F. Scioscia, S. Ieva, G. Loseto, and E. Di Sciascio. Semantic annotation of openstreetmap points of interest for mobile discovery and navigation. In *2012 IEEE First International Conference on Mobile Services*, pages 33–39, 2012. 1

[42] K. Sakurada and T. Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. In *BMVC*, 2015. 3

[43] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE CVPR*, 2015. 3, 6

[44] H. Su, S. Gong, X. Zhu, et al. Weblogo-2m: Scalable logo detection by deep learning from the web. In *ICCV Workshop on Web-scale Vision and Social Media*, 2018. 3, 8

[45] H. Su, X. Zhu, and S. Gong. Deep learning logo detection with data expansion by synthesising context. In *IEEE WACV*, 2017. 3, 8

[46] H. Su, X. Zhu, and S. Gong. Open logo detection challenge. In *BMVC*, 2018. 7, 8

[47] G. Tolias, R. Sicre, and H. Jégou. Particular object retrieval with integral max-pooling of CNN activations. In *ICLR*, 2016. 7

[48] A. Torii, J. Sivic, M. Okutomi, and T. Pajdla. Visual place recognition with repetitive structures. In *IEEE CVPR*, 2013. 2, 3

[49] A. Tüzkö, C. Herrmann, D. Manger, and J. Beyerer. Open Set Logo Detection and Retrieval. In *VISAPP*, 2018. 3, 8

[50] E. Ustinova and V. Lempitsky. Learning deep embeddings with histogram loss. In *NIPS*, 2016. 3, 6, 7

[51] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *IEEE CVPR*, 2016. 3, 5

[52] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *IEEE CVPR*, 2014. 3

[53] Y. Wang, Z. Luo, and P.-M. Jodoin. Interactive deep learning method for segmenting moving objects. *Pattern Recognition Letters*, 96:66 – 75, 2017. Scene Background Modeling and Initialization. 3

[54] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. SUN Database: Large-scale scene recognition from abbey to zoo. In *IEEE CVPR*, 2010. 2

[55] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017. 2

[56] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, 2014. 2