Street-View Change Detection with Deconvolutional Networks

Pablo F. Alcantarilla[1*], Simon Stent[2*], German Ros[3], Roberto Arroyo[4] and Riccardo Gherardi[5]
[1]iRobot Corporation, London, UK – palcantarilla@irobot.com
[2]Department of Engineering, University of Cambridge, Cambridge, UK – sistent@cantab.net
[3]Computer Vision Center, UAB, Barcelona, Spain – gros@cvc.uab.es
[4]Department of Electronics, University of Alcalá, Madrid, Spain – roberto.arroyo@depeca.uah.es
[5]Toshiba Research Europe Ltd., Cambridge, UK – riccardo.gherardi@crl.toshiba.co.uk

***Abstract*—We propose a system for performing structural change detection in street-view videos captured by a vehicle-mounted monocular camera over time. Our approach is motivated by the need for more frequent and efficient updates in the large-scale maps used in autonomous vehicle navigation. Our method chains a multi-sensor fusion SLAM and fast dense 3D reconstruction pipeline, which provide coarsely registered image pairs to a deep deconvolutional network for pixel-wise change detection. To train and evaluate our network we introduce a new urban change detection dataset which is an order of magnitude larger than existing datasets and contains challenging changes due to seasonal and lighting variations. Our method outperforms existing literature on this dataset, which we make available to the community, and an existing panoramic change detection dataset, demonstrating its wide applicability.**

I. Introduction

When viewed at the scale of cities and over periods spanning seasons or years, the urban visual landscape is a highly dynamic environment, with many navigational landmarks such as buildings, traffic signs and other road-side structures being constantly added or removed [24, 25, 34]. From the viewpoint of an autonomous driving system, maintaining an up-to-date map of such landmarks is essential. The higher the frequency of map updates, the more robust the system's navigation and planning is likely to be.

In this work we seek to address the problem of efficient map maintenance by means of structural change detection using a minimal sensor suite: low-quality monocular cameras, GPS and inertial odometry. In a spirit similar to [38], we believe that a cost-effective solution (obtained by removing the reliance on LiDAR sensors) could be more widely adopted and lead to quicker, distributed map updates through crowdsourcing. Up-to-date maps are not only useful for robust navigation, but may also yield other benefits such as monitoring the availability of parking spaces or route closures and diversions due to temporary roadworks.

Detecting structural changes in images of road scenes from monocular images is a challenging problem, as illustrated in Fig. 1. Images taken at different times exhibit large variability that may be induced by changes of interest, such as structural changes (construction, building demolition, traffic signs), but also from nuisances such as viewpoint changes, outdoor conditions (lighting, weather, season) and dynamic changes (pedestrians, vehicles, vegetation). In order to successfully differentiate between structural changes and nuisances, the detection method must be capable of modelling both.

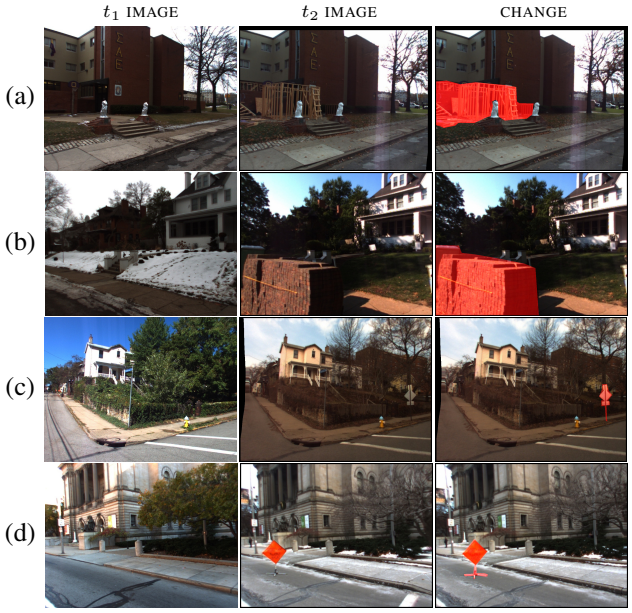

Fig. 1. Challenging examples of street-view change detection taken from our dataset. Left and middle columns show registered images from time 1 and time 2, right column shows ground truth structural changes highlighted in red. Note that all examples exhibit significant changes in lighting, weather and season. In sequence: (a) a new construction appears on the outside of an existing building; (b) construction materials are left by the side of the road; (c) a new road signal is installed; (d) a temporary traffic sign blocks the road.

Our proposed system is summarised in Fig. 2. We first register video sequences from different times using an end-to-end multi-sensor fusion Simultaneous Localization and Mapping (SLAM) system in combination with an efficient dense 3D reconstruction system. The SLAM system estimates vehicle trajectory and a sparse 3D scene reconstruction by fusing information from GPS, inertial odometry and cameras. Registration across seasons is achieved by approximate GPS localization followed by robust feature-matching and Bundle Adjustment (BA). Scene reconstructions from different time instances are densified using an efficient slanted plane

Acknowledgements: The authors would like to thank Toshiba Research Europe for their support in the project, the conference reviewers for their helpful feedback, the Spanish MEC Project TRA2014-57088-C2-1-R and NVIDIA Corporation for generous support of hardware.

*First two authors contributed equally.

smoother approach which is able to cope with the challenge of untextured or poorly illuminated regions that are commonly found in urban scenes. The combination of dense geometry and accurate registration allows images from different times to be warped into alignment with one another for comparison, mitigating a key source of nuisance variation caused by changing viewpoints.

Next, inspired by the recent success of deep Convolutional Neural Networks (CNNs) at learning image invariances for large-scale image processing tasks (e.g. image classification [20], semantic segmentation [21, 28] and place recognition [37]), we adopt a deep deconvolutional architecture [31] for the task of structural change detection. Our network, illustrated in Fig. 3, takes as input the aligned image pairs and returns a pixel-wise classification of structural changes.

We demonstrate through experimentation on an extensive, manually labelled test dataset that our method is able to predict structural changes with better performance than existing approaches. Our dataset is derived from a subset of the *Visual Localization CMU* (VL-CMU) dataset[1], originally created for long-term topometric visual localization [2, 3]. We generate 1,362 registered image pairs (containing 152 sequences of real changes with corresponding ground truth annotations), taken of the city of Pittsburgh, PA, USA, over a period of one year.

The main contributions of our paper are as follows:

1) We propose a deep deconvolutional architecture that significantly outperforms hand-crafted descriptors [18, 23, 40] and a CNN-based approach [32] on the task of street-view change detection while remaining suitably lightweight for embedded devices (1.4M parameters).
2) We introduce a novel dataset for the task of change detection in urban scenarios that is an order of magnitude larger than existing datasets and contains challenging seasonal and lighting variations. We make this dataset available for public download and use at our project website: http://www.saistent.com/proj/RSS2016.html.
3) To create our dataset we have designed a multi-sensor fusion SLAM system coupled with a fast dense reconstruction pipeline for the approximate alignment of image pairs for change detection across time.

II. Related Work

Image-based change detection [30] is an important problem in computer vision and robotics, since identifying the areas of change in a scene is the first step towards many common tasks. For example, it can improve the efficiency of 3D map maintenance by restricting updates to only the changed areas [38, 41] or allow a system to learn about the nature of objects in the environment by segmenting them as they change [7].

In urban change detection, which is the focus of our work, the typical change detection pipeline separates into two parts: registration and similarity computation. Registration is needed because for typical large urban datasets, images are often sparsely sampled and it is common to have a baseline of several metres when comparing one image to the closest matching picture across time (resulting in significant perspective deformations). Registration is typically addressed using Structure from Motion (SfM), followed by either multi-view stereo techniques or model-based reconstruction, which allow the generation of surface models that can be used to warp images from one camera viewpoint to another via reprojection [25, 36, 38].

Generating sufficiently detailed surface models in the wild is a challenging problem in itself and several works propose alternative solutions. Where cadastral city models are available, [39] show that they can be used in place of multi-view stereo. Where more precise, centimetre scale range measurements from LiDAR sensors are available, methods such as [29] show that precision can be much improved. The work by [33] defines a probabilistic framework in which changes over all possible disparities are evaluated and integrated for each reprojected ray. While this avoids the need for explicit modelling and additional information or sensors, their framework makes the assumption of per-pixel independence in order to be tractable but still remains computationally expensive.

In our work, we seek a computationally efficient approach which works from image data, GPS and inertial odometry on general unstructured scenes. We propose a novel dense reconstruction technique which copes especially well with the untextured or poorly illuminated regions that challenge existing multi-view stereo (MVS) methods [8, 11].

After the image sequences have been aligned through registration and projection, a similarity measure is employed to determine *changes of interest* between the data while ignoring other *nuisance changes*. The definition of what constitutes a change of interest or a nuisance change varies depending on the task. Changes of interest may be purely geometric, such as the appearance or disappearance of urban structures [33, 38, 39], or textural, such as changes in billboards or shop-fronts [25] or surface defects [36]. Prototypical nuisance changes which the similarity measure must be invariant to include vegetation (e.g. a tree changing from green to red in the autumn, then shedding its leaves in the winter) and lighting effects (e.g. cast shadows or underexposed images).

The similarity functions are typically a combination of absolute differences of color, depth and distances between hand-crafted descriptors. Recently pre-trained deep convolutional networks were proposed to detect changes in urban scenarios [32]. Despite having reasonable performance, the method relies on superpixel regularization and sky/ground segmentation to delineate changes accurately. Other works such as [36] propose training change detection networks from scratch on image patches to classify changes for industrial inspection. In contrast to these prior works, we adopt a deconvolutional network approach [28, 31] and demonstrate its ability to learn an appropriate, spatially precise similarity function for this challenging outdoor problem.

[1]Available from: http://3dvis.ri.cmu.edu/data-sets/localization/

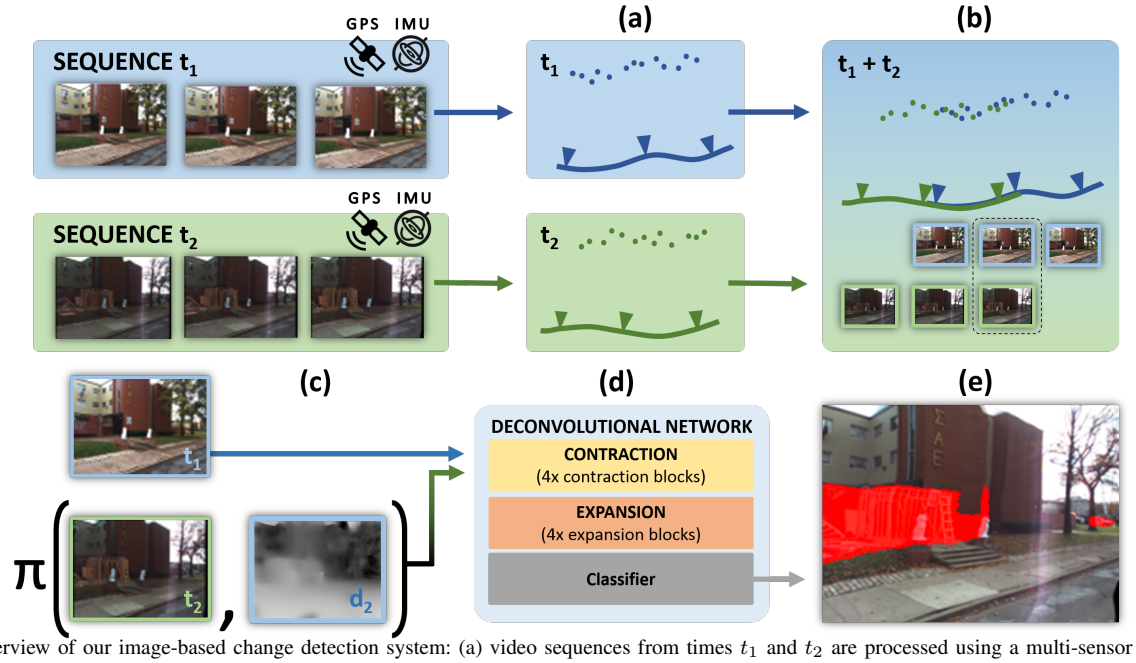

Fig. 2. Overview of our image-based change detection system: (a) video sequences from times t_1 and t_2 are processed using a multi-sensor fusion SLAM system, taking into account GPS, inertial odometry and RGB image data, to yield vehicle motion and sparse 3D reconstructions; (b) sequences are registered across time via approximate GPS localization followed by robust feature-matching and Bundle Adjustment; (c) reconstructions are efficiently densified using a novel slanted plane smoother approach; depth maps are used to align the image by reprojection (π); (d) a deconvolutional network is used to predict changes between aligned RGB images; (e) the predicted changes of our network are shown in red. Nuisances due to lighting and seasonal change are correctly handled.

III. Deconvolutional Networks for Change Detection

We propose to detect changes between pairs of images using an efficient CNN architecture based on the idea of stacking contraction and expansion blocks [22, 28, 31]. Our network, which we refer to as CDNet, takes its design from [31], consisting of 1.4 million parameters, making it very compact compared to 134 millions for FCN-8s [22]. We use this architecture as it was found to offer a good trade off between performance and model size, being small enough to be suitable for a mobile application and not prone to overfitting on small datasets.

A. Net Definition

A graphical summary of CDNet is shown in Fig. 3. The four blocks forming the contraction stage serve to create a rich representation that allows for recognition as in standard classification CNNs. The four blocks forming the expansion stage are used to improve the localization and delineation of changed regions. The final change decision is made by a softmax linear classifier operating densely per pixel.

Each contraction block consists of a 7×7 convolution layer with a fixed number of 64 features. Its outputs are normalized by using batch normalization prior to non-linear activation, in order to reduce the internal covariate shift [16] during training and improve convergence. In our case, batch normalization parameters are not computed using statistics but instead learned as extra parameters. This allows batch normalization to be easily bypassed at test time. Non-linear activations are produced by standard Rectified Linear Units (ReLU) [10], and are proceeded by a 2×2 max-pooling layer with stride 2 to reduce each spatial dimension by 2. During this operation the indices of the maximum responses are stored for later use in the corresponding expansion block to perform a clean upsampling of the data. This is necessary to produce sharp edges and avoid blocky results [28].

Each expansion block starts by upsampling its input using an unpooling layer. This layer makes use of the previously stored indices to produce an upsampled version of the input that conserves activations at the location of edges, and other high frequency features. This operation is followed by a 7×7 convolution with a fixed number of 64 features. As before, pre-activations are normalized using batch normalization before ReLU. This stack of expansion and contraction blocks makes the network architecture fully symmetric in terms of numbers of features. No fully-connected layers are used in our architecture, giving rise to very efficient models suitable for mobile applications. For further details, refer to [31].

B. Training Approach

Both convolution and deconvolution blocks are randomly initialized using He's method [14]. Training was carried out using the Adam optimizer [19] with default parameters. This choice resulted in faster training convergence than standard stochastic gradient descent, converging within 200 epochs, with 150 batches per epoch and a batch size of 10 image pairs. The use of weighted cross-entropy as the loss function, with weights chosen according to the inverse frequencies of the classes in the training set, was also found to be a necessary ingredient to achieve the results we report. The model was implemented and trained using MatConvNet [42].

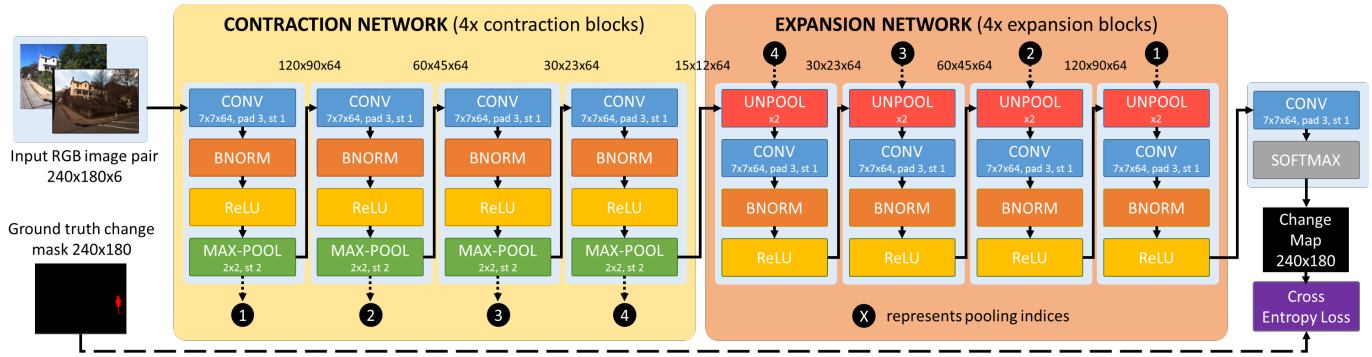

Fig. 3. We propose a deconvolutional network architecture for change detection called CDNet. Each of the four contraction blocks consists of a convolution (CONV), batch normalization (BNORM), ReLU and max-pooling layer. Each of the four expansion blocks consists of a guided unpooling (using the stored pooling indices from the corresponding contraction block), convolution, batch normalization and ReLU layer. The final layer is a linear operator followed by a soft-max classifier. Channel-wise normalisation is applied to the input RGB images as a pre-processing step.

IV. Building a Dataset of Changes Across Seasons

We exhaustively explored the VL-CMU dataset to gather a set of typical urban macroscopic changes such as those presented in Fig. 1, for training and evaluating our framework. The VL-CMU dataset was originally proposed for studying topometric localization [2, 3] and consists of 16 sequences captured over the period of one year in the city of Pittsburgh, PA, USA. The sequences are recorded at 15Hz by a pair of 1024×768 pixel Point Grey Flea 2 vehicle-mounted cameras at 45° degrees left and right from the forwards direction and zero overlap between the pair. In each of the sequences the vehicle traversed approximately the same 8km route. The dataset also includes measurements from an intertial sensor, GPS and a single line scanning SICK LiDAR. We do not make use of the latter since the LiDAR's external calibration parameters with respect to the other sensors are not provided within the dataset.

From the VL-CMU dataset we identified and extracted 152 RGB and depth image sequences for change detection. Each sequence contains on average 9 pairs of corresponding images taken from different time instances. These are used to generate a total of $1,362$ registered image pairs, each with a manually annotated ground truth structural change mask and sky mask.

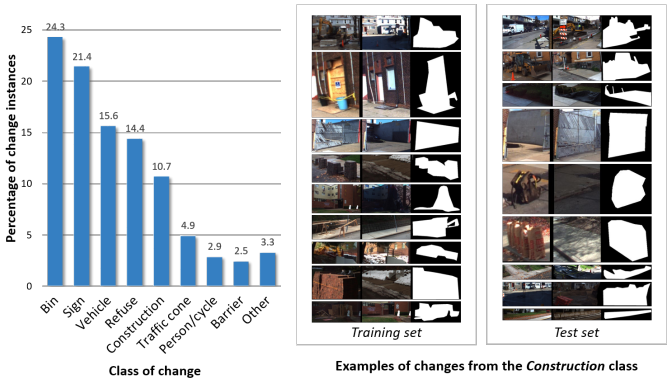

Fig. 4. Left: distribution of changes by class in the VL-CMU-CD dataset. Right: example changes in the training and test data for the category of *Signs*. Each change consists of a query and reference image and black and white change mask.

Our dataset, which we name VL-CMU-CD, is compared against existing datasets in Table I. It is an order of magnitude larger than existing datasets. While Taneja et al. [39] use a dataset of 1,000 panoramic images for street-view change detection, we exclude them from the table as they compare images against a cadastral 3D model rather than image-to-image pairs. In addition to its larger size, since VL-CMU-CD is captured over the course of a year, it contains challenging natural seasonal changes, varying environmental and meteorological conditions, structural changes such as new buildings, construction areas or changes in road signalling, and is more representative of the typical types of changes that can be observed in an urban environment than the perspective datasets of [33, 38]. The dataset of Sakurada and Okatani [32] is similar in this regard, but the image pairs are not sequential and do not allow the recovery of depth for image alignment. In Fig. 4 we provide a breakdown of the changes in VL-CMU-CD by class and a selection of the changes for the *Construction* class in testing and training datasets. The five most common classes are bins (24%), signs/traffic-signs (21%), vehicles (16%), refuse (14%) and construction/maintenance work (11%).

TABLE I
Comparison of existing street-view change detection datasets.

Dataset	# Seq.	# Pairs	Type
Taneja et al. [38]	4	~50	perspective
Sakurada et al. [33]	23	92	perspective
Sakurada and Okatani [32]	-	200	panoramic
VL-CMU-CD (Ours)	152	1,362	perspective

V. Image Alignment via Multi-Sensor Fusion SLAM and Dense 3D Reconstruction

In this section we explain in sequence our multi-sensor fusion SLAM system for camera trajectory and sparse 3D map estimation, our approach to sequence-to-sequence registration across time and season, and our dense reconstruction framework using a slanted plane smoother approach.

A. Multi-Sensor Fusion SLAM

We formulate the multi-sensor fusion SLAM problem as a factor graph, in which each factor encodes the connectivity

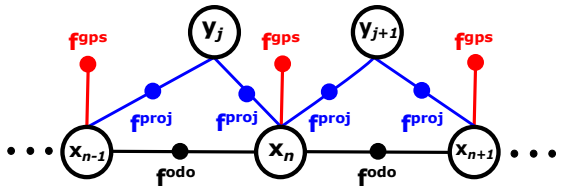

Fig. 5. A simplified example of a factor graph encoding the relationship between camera poses and scene structure in multi-sensor fusion SLAM.

between the unknown variable nodes and the sensor measurements. The final goal is to estimate a set of M camera poses $X = \{x_i\}_{i=1:M}$ and N reconstructed 3D points $Y = \{y_j\}_{j=1:N}$, given a set of sensor measurements, Z. In our formulation we assume Z to consist of 2D image feature measurements, z_{ij}^{proj}, a GPS measurement per camera pose, z_i^{gps}, and an equivalent odometry factor (composition of multiple odometry measurements) between any two camera frames, z_{i_1,i_2}^{odo}. More detailed information about multi-sensor fusion SLAM can be found in [15].

The joint probability distribution of the navigation variables $\Theta = \{X^*, Y^*\}$ given the measurements Z, can be factorized as the product of the contribution of each individual factor in the graph:

$$P(X,Y;Z) \propto \prod_{k=1}^{K} f_k\left(\Theta_v^k\right), \tag{1}$$

where Θ_v^k represents a subset of the variable nodes and K is the total number of factors in the graph. Each factor f_k represents an error function that connects a set of variables and measurements. We assume Gaussian noise distributions for all factors. Fig. 5 illustrates a simplified example of a typical factor graph for the task. The factor formulations for each sensor are as follows:

*1) **Projection factor**:* measures the reprojection error of a 3D point $y_j \in \mathbb{R}^3$ for a particular camera pose $x_i \in \mathrm{SE}(3)$ given the corresponding 2D image measurement $z_{ij}^{proj} \in \mathbb{R}^2$:

$$f^{proj}\left(x_i, y_j\right) = \exp\left(-\frac{1}{2}\|\pi\left(x_i, y_j\right) - z_{ij}^{proj}\|_{\Sigma_{ij}}^2\right), \tag{2}$$

where $\pi\left(x_i, y_j\right)$ is the projection of y_j into x_i assuming known camera intrinsics and $\|\cdot\|_{\Sigma_{ij}}^2$ is the squared Mahalanobis distance induced by the measurement covariance Σ_{ij}. In order to deal with outliers and spurious measurements, we use the pseudo-Huber loss function as a robust kernel [13].

*2) **GPS factor**:* a unary factor that enforces a constraint in the camera position:

$$f^{gps}\left(x_i\right) = \exp\left(-\frac{1}{2}\|h^{gps}(x_i) - z_i^{gps}\|_{\Sigma_i}^2\right), \tag{3}$$

where $h^{gps}(x_i)$ models the GPS measurement function.

*3) **Odometry factor**:* a binary factor measuring the relative motion between two camera poses:

$$f^{odo}\left(x_{i_1}, x_{i_2}\right) = \exp\left(-\frac{1}{2}\|h^{odo}(x_{i_1}, x_{i_2}) - z_{i_1 i_2}^{odo}\|_{\Sigma_{i_1 i_2}}^2\right), \tag{4}$$

where $h^{odo}(x_{i_1}, x_{i_2})$ models the relative transformation between two camera poses.

We extract and track features for each frame using A-KAZE features [1], due to its high repeatability and density of detected features. Features are tracked by finding matches within confidence regions predicted by odometry measurements.

Calculating the *maximum-a-posteriori* (MAP) estimate of Eq. (1) is equivalent to minimizing a weighted non-linear least squares function using the contribution of each factor in the graph. Our optimization uses a Local Bundle Adjustment (LBA) formulation [27]. In LBA, the estimated parameters are the poses of the m most recent keyframes and 3D points which have at least one observation in these frames, considering only the reprojection errors from the M_{LBA} most recent keyframes. In all experiments we set m=3 and M_{LBA}=10.

B. Registering Sequences Across Seasons

The use of GPS in Section V-A provides an approximate registration between two sequences of images from times t_1 and t_2. However, GPS errors can often be up to 10 m in urban areas and do not provide the necessary accuracy for the task of change detection; as a consequence we refine the registration by robustly matching A-KAZE descriptors across image sets via RANSAC and a three-point algorithm. Once the set of common correspondences is identified, we perform global BA and refine the set of cameras poses and sparse 3D point clouds to a common reference frame. This worked well for our experiments, but in practice may be sensitive to (i) the length of sequences being matched – shorter sequences may be harder to match if visual changes are very significant (ii) the spacing between cameras (speed of the vehicle and image capture rate) and (iii) outliers in GPS readings. Approaches such as [4] to train features to match across dynamic lighting conditions could further improve the registration in challenging scenarios.

C. Dense 3D Reconstruction

We perform dense 3D reconstruction to generate surface models that can be used to warp images from one camera viewpoint to another via reprojection. Our dense 3D reconstruction algorithm is partially inspired by the Efficient Large-Scale Stereo Matching (ELAS) method [9]. ELAS is a stereo matching method that uses support points (*i.e.* points in textured regions that can be matched reliably) to guide matching in surrounding pixels via a Delaunay triangulation in the image space using a planarity smoothness term.

One of the drawbacks of ELAS is that it assumes the availability of a relatively uniform distribution of support points, to ensure that the set of hypothesized planes resulting from triangulation exhibit a regular size. This assumption is often broken in urban scenarios, in which there are many untextured or poorly illuminated regions that challenge existing multi-view stereo approaches [8, 11, 17].

We address this drawback by extending ELAS in a coarse-to-fine fashion and replacing the Delaunay triangulation by superpixels, as summarised in Alg. 1. For an image I_i with pose x_i, we start at the coarsest octave of scale space, O_{max}, and estimate a depth map d. To do so, we first assign a set of

visible 3D points Y_i^{vis} as support points. The set is chosen by selecting features whose scale lies in the same coarsest octave of scale space.

We then extract a set of SEEDS superpixels [6], with size proportional to the scaled image, and fit a plane for the support points belonging to each superpixel via RANSAC. For each support point, we use its 2D measurement and depth estimate as inputs for plane fitting.

For those superpixels in which we have enough support points to fit a plane (*i.e.* at least three non-collinear points), we minimize an energy function based on a data term and two smoothing terms that penalize deviations between the depth candidate for a particular pixel d_j, the depth planar prior d_{pl} and the prior depth estimate from the previous scale level d_{pr}:

$$E(d) \;=\; E_{data}(d_j, \hat{d}_j) + \lambda_1 E_s(d_j, d_{pl}) + \lambda_2 E_s(d_j, d_{pr}), \quad (5)$$

where λ_1 and λ_2 control the contribution of the smoothing terms E_s in the overall energy. As data term, we use the difference between DAISY descriptors [40] computed for each pixel in the reference image I_i and its projection in the closest image I_{i+1} using the camera parameters, a hypothesized depth d_j and its projection $\hat{d}_j$. For the smoothing terms, we adopt Gaussian functions. For superpixels where it is possible to fit a plane, we evaluate the energy in Eq. (5) for each pixel in the reference image using a small range of depth hypotheses centered at the prior value. At the coarsest scale level, for those superpixels where it is not possible to fit a plane, we explore the full range of depth values $(d_{min}, \ldots, d_{max})$ without any smoothing terms. The depth estimate is refined by means of a fast global smoother (FGS) based on weighted least squares as described in [26]. The solution is then propagated to the next scale level, repeating until the desired resolution is reached.

While the algorithm has been described for clarity of exposition on a pair of consecutive views $(I_i,\ I_{i+1})$, it can be trivially extended to multiple views by expansion of the data term. Fig. 6 shows the output of our method on a test image, with support points, superpixels and estimated depth map using three scale levels.

Algorithm 1 Dense 3D Reconstruction from Video Sequences

Input Images I_i, I_{i+1}, camera poses x_i, x_{i+1}, set of visible 3D points Y_i^{vis}, number of octaves O_{max}
Output Estimated depth map d for image I_i

for $j = O_{max} \rightarrow 1$ **do**
1. Compute scale factor ratio $\sigma = 2^j$; adjust calibration parameters
2. Compute DAISY descriptors for I_i^j and I_{i+1}^j
3. Compute SEEDS superpixels for images I_i^j and I_{i+1}^j.
4. For each superpixel, estimate a plane via RANSAC
5. For each pixel in each superpixel, minimize (5)
6. Use FGS to smooth the depth estimate using image I_i^j from the scale space as guide image in the smoother
7. Propagate solution to next scale level
end for

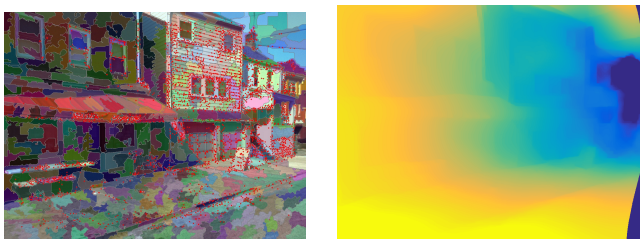

Fig. 6. Left: extracted superpixels with projected support points in red. Right: estimated depth map. Best viewed on screen.

VI. Experimental Results

In this section, we first describe the suitability of our dense reconstruction method for the task of image alignment, compared to the output of [8] utilized by [38]. We then examine the performance of our proposed deconvolutional network for change detection versus existing approaches using both our VL-CMU-CD dataset and the recently published Tsunami and Google Street View (GSV) datasets from [32].

A. Dense 3D Reconstruction

The goal of our dense 3D reconstruction pipeline was to efficiently produce dense depth maps with sufficient accuracy for the alignment of images for change detection. We compare our dense 3D reconstruction with respect to PMVS [8] used by [38] on a random image sequence from our dataset.

Since there is no ground truth information available for dense 3D geometry in VL-CMU, we perform a quantitative comparison showing the number of reconstructed 3D points and density of pixels containing a depth estimate as quality measures. For our method, we fused individual depth maps considering geometric consistency between three consecutive views. Fig. 7 depicts a close-up view from the renderings of the 3D point clouds of the two methods. Table II details the pixel density for which we obtain a depth estimate and the total number of reconstructed 3D points. Our method obtains a much higher density and number of 3D points than PMVS, which struggles on textureless regions. A high density allows accurate image warping from one camera viewpoint to another via reprojection without resorting to time-consuming meshing methods. On average our method takes about 2 seconds for processing a pair of images on a single threaded implementation on a 3 GHz desktop PC and may be accelerated by parallelization. We include a video as supplementary material to illustrate the quality of our reconstruction.

One of the key differences between our approach and existing literature is that we do not require the concatenation of several sequential batch steps. Methods like [35, 38] first employ an SfM pipeline to obtain camera positions followed by PMVS and promotion of the resulting point cloud to a mesh. For example, Poisson surface reconstruction is used in [38] to obtain a denser surface model from the original PMVS point cloud. The same approach is employed in [35] but coupled with an edge-preserving smoother and information from occluding contours. In [5], the sparse point cloud from PMVS is used as an input for local warping of superpixels for view synthesis. Such methods are batch by nature and

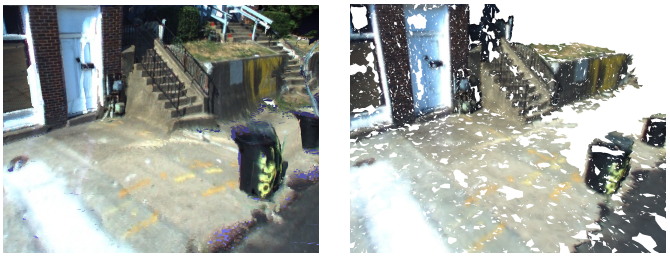

Fig. 7. Qualitative comparison of dense 3D reconstruction. Left: Our method. Right: PMVS results after ball pivoting surface reconstruction. Our approach is denser and one order of magnitude faster to compute (26×).

TABLE II
DEPTH RECONSTRUCTION COMPARISON FOR A SEQUENCE OF 43 FRAMES.

Method	Density %	# 3D Points	Processing Time (s)
PMVS	25	217,703	2,666
Ours	77	3,449,148	102

comparatively more computationally expensive. Our proposed method may in future be incorporated into incremental reconstruction pipelines and is well suited to real-time processing via substantial code optimisation and parallel processing.

B. VL-CMU-CD Dataset

To evaluate CDNet on our VL-CMU-CD dataset, we divided it into a training set of 933 image pairs (98 sequences) and a test set of 429 image pairs (54 sequences). The split between the sequences was chosen at random, with whole image descriptor matching used to confirm that test and training sets did not contain similar looking sequences. We used a small subset of the training data for validation to tune convergence for optimization. For final experiments, we trained on the full training set without validation, performing early stopping after 150 epochs in all cases to ensure fairness. In addition, we used data augmentation to help prevent overfitting during training, by adding image pairs containing both artificial changes (by adding synthetic changes to existing images) and no changes of interest, in the manner of [36]. These were added in the approximate ratio of 35% real changes to 65% augmented changes.

We compared CDNet against multiple baselines including depth only (obtained with our dense 3D reconstruction pipeline), hand-crafted descriptors such as dense SIFT [23], DAISY [40], DASC [18]), and a pre-trained CNN for image recognition combined with superpixel regularization, as described in [32]. For the depth-only approach we compare raw absolute difference between depth maps normalised by the larger of the two depths per pixel.

Quantitative Comparison. Fig. 8 depicts a quantitative comparison of our method's test set performance on VL-CMU-CD versus baseline methods. We show both *False Positive Rate* (FPR) vs. *True Positive Rate* (TPR or Recall) and *Precision-Recall* graphs. Table III compares our method over different change detection metrics [12] for an FPR of 0.10 and 0.01. Our deconvolutional network outperforms other methods on all metrics by a significant margin.

Qualitative Comparison. Fig. 9 illustrates the predicted

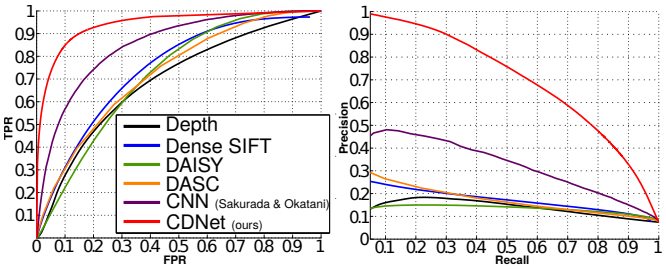

Fig. 8. VL-CMU test results: (a) FPR vs TPR curves. (b) Recall vs Precision curves.

TABLE III
QUANTITATIVE COMPARISON VS BASELINE METHODS AT $FPR = 0.10$ AND 0.01. BEST RESULTS ARE HIGHLIGHTED IN BOLD.

	$FPR = 0.10$			$FPR = 0.01$		
Method	Pr	Re	F_1	Pr	Re	F_1
Depth	0.18	0.28	0.22	0.08	0.01	0.02
D-SIFT [23]	0.20	0.31	0.24	0.25	0.04	0.07
DAISY [40]	0.15	0.22	0.18	0.11	0.02	0.03
DASC [18]	0.20	0.30	0.23	0.29	0.05	0.08
CNN [32]	0.31	0.57	0.40	0.48	0.12	0.18
CDNet (ours)	**0.40**	**0.85**	**0.55**	**0.79**	**0.46**	**0.58**

change maps of several of the methods for randomly selected test image pairs. The performance of our network is markedly better than other methods; in the majority of cases it is able to locate and delineate the changes of interest. The closest performing method of [32] is relatively good at identifying change, but misses numerous key changes (the post in S2, truck in S4, sign in S6, fence in S7), and delineation of the changes is rather poor despite the use of superpixels.

C. Tsunami and Google Street View Datasets

Additionally, we evaluate the performance of CDNet on the panoramic change detection datasets of [32]. Because CDNet is "fully convolutional", we apply it directly to the 224×1024 pixel images of the dataset without resizing. However, because the data is from a very different domain to VL-CMU-CD, captured using a panoramic camera, we first fine-tune CDNet over 30 epochs on a training set extracted from the data.

Table IV reports our average F_1 score for each dataset trained using 5-fold cross validation on 80-20 training-validation splits. We improve on the method of [32] for the Tsunami dataset, despite using a single end-to-end network, without superpixel smoothing or sky segmentation, and using one order of magnitude fewer parameters (1.4 million vs 14.7 million). However, our performance on the GSV dataset is slightly worse. Close inspection of the dataset revealed that registration errors between image pairs are more significant in this latter, more sparsely captured dataset. It is likely that our network cannot achieve translation invariance as effectively as the much deeper net used by [32] but conversely, when registration is better, our model is more discriminative despite its small relative size. This justifies the use of efficient dense

Fig. 9. Illustration of change detection performance of our method versus existing methods on a variety of challenging sequences of the test dataset. Images exhibit significant changes in lighting, weather and season. Changes detected are labelled in red. From left to right: (i) reference image; (ii) registered query image; change detection results at 10% false positive rate using (iii) Dense SIFT [23], (iv) CNN [32], (v) Depth, (vi) proposed approach; (vii) ground truth.

reconstruction for image alignment in our approach.

TABLE IV
F_1 SCORE FOR TSUNAMI AND GOOGLE STREET VIEW DATASETS [32].

Method	Tsunami	GSV
Dense SIFT	0.649	0.528
Sakurada and Okatani [32]	0.724	**0.639**
CDNet (ours)	**0.774**	0.614

VII. CONCLUSIONS AND FUTURE WORK

We have proposed and evaluated a novel approach to street-view change detection from monocular video sequences. Our method combines geometric methods (SLAM and dense 3D reconstruction, used for the approximate registration of video sequences) with the learning of an efficient deconvolutional network to discriminate between actual changes and nuisance changes such as caused by lighting and seasonal variation.

Our approach has been shown to outperform existing approaches on a novel, large and challenging change detection dataset which we make available for public download and use at our project website. We have additionally shown the effectiveness of the deconvolutional network on an existing benchmark dataset.

We are aware that our dataset, despite being significantly larger than currently available collections, remains some distance from the size of classification datasets such as ImageNet and can be scaled up much further. We are convinced that an approach such as ours will scale effectively to larger datasets as they become available. Two promising directions for scaling such change detection datasets are to leverage dense LiDAR scans to reduce the need for manual annotation or to generate realistic synthetic data.

References

[1] P. F. Alcantarilla, J. Nuevo, and A. Bartoli. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. In *British Machine Vision Conf. (BMVC)*, 2013.

[2] H. Badino, D. Huber, and T. Kanade. Visual Topometric Localization. In *IEEE Intelligent Vehicles Symposium (IV)*, 2011.

[3] H. Badino, D. Huber, and T. Kanade. Real-Time Topometric Localization. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2012.

[4] N. Carlevaris-Blanco and R. M. Eustice. Learning Visual Feature Descriptors for Dynamic Lighting Conditions. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014.

[5] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis. Depth Synthesis and Local Warps for Plausible Image-based Navigation. *ACM Transactions on Graphics*, 32, 2013.

[6] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani, and L. Van Gool. SEEDS: Superpixels Extracted via Energy-Driven Sampling. In *Eur. Conf. on Computer Vision (ECCV)*, 2012.

[7] R. Finman, T. Whelan, M. Kaess, and J. J. Leonard. Toward Lifelong Object Segmentation from Change Detection in Dense RGB-D Maps. In *European Conf. on Mobile Robots (ECMR)*, 2013.

[8] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *IEEE Trans. Pattern Anal. Machine Intell.*, 32(8):1362–1376, 2010.

[9] A. Geiger, M. Roser, and R. Urtasun. Efficient Large-Scale Stereo Matching. In *Asian Conf. on Computer Vision (ACCV)*, pages 25–38, 2010.

[10] X. Glorot, A. Bordes, and Y. Bengio. Deep Sparse Rectifier Neural Networks. In *Intl. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 315–323, 2011.

[11] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S.M. Seitz. Multi-View Stereo for Community Photo Collections. In *Intl. Conf. on Computer Vision (ICCV)*, pages 1–8, 2007.

[12] N. Goyette, P-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. Changedetection. net: A new change detection benchmark dataset. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012.

[13] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Intl. Conf. on Computer Vision (ICCV)*, 2015.

[15] V. Indelman, S. Williams, M. Kaess, and F. Dellaert. Information fusion in navigation systems via factor graph based incremental smoothing. *Journal of Robotics and Autonomous Systems*, 61(8):721–738, 2013.

[16] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Intl. Conf. on Machine Learning (ICML)*, 2015.

[17] Z. Kang and G. Medioni. 3D Urban Reconstruction from Wide Area Aerial Surveillance Video. In *Workshop on Applications for Aerial Video Exploitation (WAVE)*, 2015.

[18] S. Kim, D. Min, B. Ham, S. Ryu, M. N. Do, and K. Sohn. DASC: Dense Adaptive Self-Correlation Descriptor for Multi-modal and Multi-spectral Correspondence. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[19] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *Intl. Conf. on Learning Representations (ICLR)*, 2015.

[20] A. Krizhevsky, I. Sustkever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[21] M.-Y. Liu, S. Lin, S. Ramalingam, and O. Tuzel. Layered Interpretation of Street View Images. In *Robotics: Science and Systems (RSS)*, 2015.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[23] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Intl. J. of Computer Vision*, 60(2): 91–110, 2004.

[24] R. Martin-Brualla, D. Gallup, and S. M. Seitz. Time-lapse mining from internet photos. *ACM Transactions on Graphics (TOG)*, 34(4):62, 2015.

[25] K. Matzen and N. Snavely. Scene Chronology. In *Eur. Conf. on Computer Vision (ECCV)*, 2014.

[26] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, and M.N. Do. Fast global image smoothing based on weighted least squares. *IEEE Trans. Image Processing*, 23(12): 5638–53, 2014.

[27] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Generic and real-time structure from motion using local bundle adjustment. 27:1178–1193, 2009.

[28] H. Noh, S. Hong, and B. Han. Learning Deconvolution Network for Semantic Segmentation. In *Intl. Conf. on Computer Vision (ICCV)*, 2015.

[29] R. Qin and A. Gruen. 3D change detection at street level using mobile laser scanning point clouds and terrestrial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 90:23–35, 2014.

[30] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *IEEE Trans. Image Processing*, 14(3):294–307, 2005.

[31] G. Ros, S. Stent, P. F. Alcantarilla, and T. Watanabe. Training Constrained Deconvolutional Networks for Road Scene Semantic Segmentation. *arXiv preprint arXiv:1604.01545*, 2016.

[32] K. Sakurada and T. Okatani. Change Detection from a

Street Image Pair using CNN Features and Superpixel Segmentation. In *British Machine Vision Conf. (BMVC)*, 2015.

[33] 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 *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 137–144, 2013.

[34] G. Schindler and F. Dellaert. Probabilistic Temporal Inference on Reconstructed 3D Scenes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[35] Q. Shan, B. Curless, Y. Furukawa, C. Hernandez, and S. M. Seitz. Occluding Contours for Multi-View Stereo. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[36] S. Stent, R. Gherardi, B. Stenger, and R. Cipolla. Detecting Change for Multi-View, Long-Term Surface Inspection. In *British Machine Vision Conf. (BMVC)*, 2015.

[37] N. Sünderhauf, S. Shirazi, A. Jacobson, D. Ferab, E. Pepperell, B. Upcroft, and M. Milford. Place Recognition with ConvNet Landmarks: Viewpoint-Robust, Condition-Robust, Training-Free. In *Robotics: Science and Systems (RSS)*, 2015.

[38] A. Taneja, L. Ballan, and M. Pollefeys. Image Based Detection of Geometric Changes in Urban Environments. In *Intl. Conf. on Computer Vision (ICCV)*, pages 2336–2343, 2011.

[39] A. Taneja, L. Ballan, and M. Pollefeys. City-Scale Change Detection in Cadastral 3D Models using Images. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[40] E. Tola, V. Lepetit, and P. Fua. DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo . *IEEE Trans. Pattern Anal. Machine Intell.*, 32(5):815–830, 2010.

[41] A. O. Ulusoy and J. L. Mundy. Image-based 4-D reconstruction using 3-D change detection. In *Eur. Conf. on Computer Vision (ECCV)*, pages 31–45, 2014.

[42] A. Vedaldi and K. Lenc. MatConvNet – Convolutional Neural Networks for MATLAB. 2015.