The Change You Want to See

Ragav Sachdeva Andrew Zisserman

Visual Geometry Group, Dept. of Engineering Science, University of Oxford

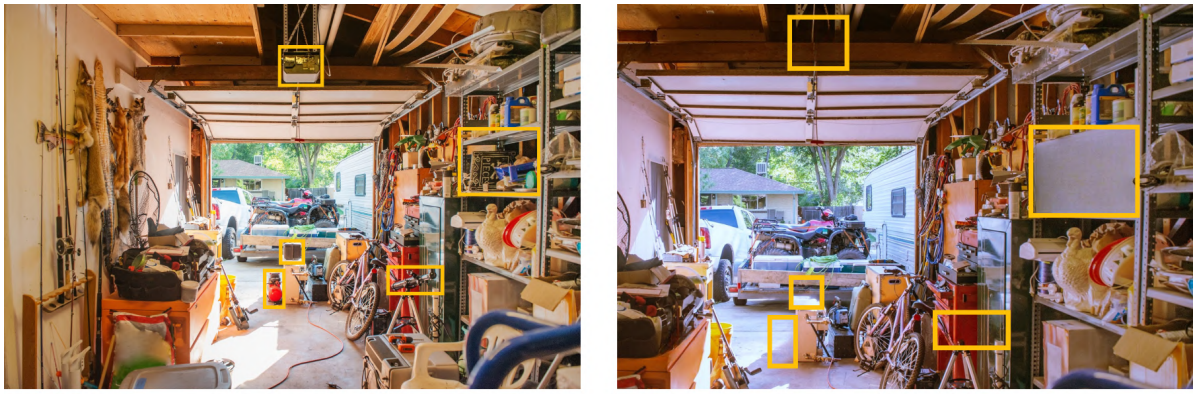


Figure 1. In this image pair, 5 out of 6 differences are shown using yellow boxes. Can you spot the remaining one? Our model can.

Abstract

We live in a dynamic world where things change all the time. Given two images of the same scene, being able to automatically detect the changes in them has practical applications in a variety of domains. In this paper, we tackle the change detection problem with the goal of detecting "object-level" changes in an image pair despite differences in their viewpoint and illumination. To this end, we make the following four contributions: (i) we propose a scalable methodology for obtaining a large-scale change detection training dataset by leveraging existing object segmentation benchmarks; (ii) we introduce a co-attention based novel architecture that is able to implicitly determine correspondences between an image pair and find changes in the form of bounding box predictions; (iii) we contribute four evaluation datasets that cover a variety of domains and transformations, including synthetic image changes, real surveillance images of a 3D scene, and synthetic 3D scenes with camera motion; (iv) we evaluate our model on these four datasets and demonstrate zero-shot and beyond training transformation generalization. The code, datasets and pre-trained model can be found at our project page: https://www.robots.ox.ac.uk/~vgg/research/cyws/.

1. Introduction

Change is all around us. Detecting changes, whether in image pairs or image sequences, is a natural computer vision task. Its applications range from the simple "spot-the-difference" game to more practical use cases such as in facility monitoring [31], medical imaging [24], satellite surveillance [9], and counterfeit detection where, for instance, a forged produce label could have subtle differences from the original.

The problem we study in this work is the following: given a pair of images, determine all the changes between them, if any. The challenge is to determine the changes between the images that are important for a particular application, while ignoring "noise" or "nuisance" variables that are irrelevant. For instance, in a surveillance application with a fixed camera, the "nuisance" parameters could be the varying lighting of the scene, changing weather conditions (e.g. rain, fog) etc. that prevent a simple "difference image" method from being applied. More generally, the two images may be taken from different viewpoints entirely so that in addition to a *photometric* transformation there maybe also be a *geometric* transformation between them. Under this setting, determining the differences can also implicitly subsume a registration problem.

We formulate this problem as the widely studied *detection* problem, wherein each change is delineated using a bounding box, as opposed to computing per-pixel changes.

This enables "object-level" change predictions and simplifies counting the number of changes between two images. To tackle this problem, we introduce a simple Siamese neural network architecture that operates on two images with geometric and photometric changes and is designed to be class-agnostic, in that it can detect changes irrespective of the object classes involved. We make use of an attention mechanism, similar to [36, 33], that can implicitly determine the correspondences between the images, register them and detect their differences.

To train this architecture, we introduce a scalable method for generating synthetic training data from real images – where for each pair of training images, we know the ground truth bounding boxes of the differences. The key idea is to leverage existing large-scale image datasets such as COCO and KITTI, and use off-the-shelf inpainting methods to inpaint various regions in an image to create differences between the inpainted and the original versions. In addition, we take measures that prevents the model from "cheating" by detecting inpainting noise. Using this dataset, we introduce both geometric and photometric transformations that we wish to be invariant to (i.e. not important for an application) by standard augmentations during training.

We demonstrate that a model trained only on this synthetic dataset using affine transformations and colour jittering can generalize in two significant ways: (i) it can be applied *zero-shot* to other datasets, and we evaluate its performance over four different datasets including different domains and both real and synthetic cases; and (ii) the transformations that it can handle extend beyond affine, and we evaluate this by including a dataset with 3D effects due to camera motion. The first generalization is a consequence of using a varied training dataset, and the second is a consequence of using attention to determine the correspondences implicitly, rather than explicitly computing geometic and photometric transformations between the images.

In summary, we make the following four contributions: (i) we introduce a novel architecture for change detection, formulated as a detection (rather than segmentation) problem that is able to implicitly learn correspondences between the images; (ii) we introduce a novel scalable method for generating a large-scale dataset of training image-pairs from existing object segmentation benchmarks; (iii) we define four evaluation datasets that cover a variety of domains and transformations: synthetic inpainted COCO image pairs related by affine transformations; a variety of images with text added in a manner consistent with the geometry of the scene; real surveillance images of a 3D scene; and synthetic 3D scenes and camera motion using the Kubric pipeline; finally, (iv) we ablate our design choices, and demonstrate zero-shot and beyond training transformation generalization.

2. Related Works

Since the notion of "change" is very broad, the problem of exploring changes in a scene has been studied under several different settings. In this section we summarise the contributions of relevant works in each category.

Change captioning: The change detection problem has been posed as a captioning problem where the model is expected to describe the differences in a pair of images in natural language. Jhamtani et al. [13] present a Spot-the-difference (STD) dataset of image pairs from surveillance cameras with text based annotations for changes and propose a method that captures visual salience by using a latent variable to align clusters of differing pixels with output sentences. Park et al. [22] focus on semantically relevant changes and present a method that performs robust change captioning on STD as well as a new change detection dataset. Oluwasanmi et al. [21] propose a fully-convolutional CaptionNet that outperforms previous methods on the STD dataset.

Street-view change segmentation: Most of the existing works that attempt to localise changes between a pair of images, formulate it as a segmentation problem, particularly in a street-view setting. Sakurada et al. [27] proposed a method for segmenting changes in a street scene using a pair of its vehicular, omnidirectional images, with the intention of detecting "city-scale" changes. Towards the same goal, Alcantarilla et al. [2] presented a system for performing structural change detection in street-view videos captured by a vehicle mounted monocular camera over time. Sakurada et al. [28] further posed a novel semantic change detection problem and proposed a weakly-supervised silhouette-based model to address it. Recently, Lei et al. [16] presented a method to locate the changed regions between a given street-view image pair and demonstrated superior results to the previous methods.

A big challenge faced by these methods is the lack of a large-scale and comprehensively labelled change dataset. Manually labelling all the changed pixels in an uncontrolled setting like streets is an extraordinarily expensive and error-prone task. TSUNAMI and GSV datasets presented in [27] contain 100 image pairs each, where the authors report spending 20 minutes to annotate each image pair. PSCD dataset presented in [28] contains 500 image pairs where the authors report spending an average of 156 minutes to annotate each image pair. Despite the remarkable effort, (a) these datasets are relatively small, and (b) their annotations are not comprehensive (by choice) e.g. changes in road signs on ground are not accounted.

Synthetic change detection datasets: An alternative to collecting and labelling real-world images is to use

synthetic datasets where the changes can be controlled. To this end, datasets such as StandardSim[1] [19] for change detection in retail stores, ChangeSim [23] for change detection in warehouses and CARLA-OBJCD[1] [11] for change detection in street scenes have been introduced. In this work, we take a tangential approach instead and leverage existing large-scale object detection datasets to train our model. Furthermore, we approach the change detection problem as a bounding box based detection problem (as opposed to segmentation) which makes it possible for us to curate various test sets, with relative ease, to reliably evaluate our model.

Change classification: Change detection has also been explored as a classification problem by Fujita et al. [7] for damage detection and Wu et al. [38] for detecting changes in book covers.

Correspondence matching: An orthogonal, yet related, problem to change detection is that of correspondence matching, where the goal is to find corresponding points in images rather than differences. There exists a plethora of literature proposing methods to find corresponding points between a pair of images [6, 15, 29, 33, 34, 35, 3, 30].

3. Architecture

Overview: Given a pair of images under some geometric transformation, our goal is to localise the changes between them in the form of bounding box predictions for *each* image. To do so, the model must have a notion of computing correspondences between the two images and establishing whether certain regions have changed, while ignoring nuisance factors such as photometric changes. Therefore, the model must simultaneously operate on both the images, coalesce their feature maps in a meaningful way and localise the changed regions.

We achieve this by first obtaining a set of dense feature descriptors for each image using a CNN-based encoder. These dense feature descriptors are then conditioned on each other using a co-attention mechanism that implicitly supplies the correspondences. Next, these conditioned feature descriptors are passed through a decoder to obtain high resolution conditioned image descriptors which are used by a bounding box detection head to localise the changes. Briefly, we employ a siamese architecture comprising of a U-Net model [25], modulated with co-attention layers [36] and concurrent Spatial and Channel Squeeze & Excitation blocks (scSE) [26], followed by a bounding box prediction head [39], as shown in Fig. 2.

In detail, given two images $I^1 \in \mathbb{R}^{3\times H\times W}$ and $I^2 \in \mathbb{R}^{3\times H\times W}$ with an unknown geometric transformation between them, the model used to localise changes between them is split into four components.

U-Net Encoder: First, we encode I^1, I^2 using a U-Net encoder (CNN), represented by $\Phi_E(\cdot)$, to obtain dense feature descriptors at multiple spatial resolutions s. Specifically, we obtain feature maps $f_s^1 \in \mathbb{R}^{c_s\times h_s\times w_s}$ and $f_s^2 \in \mathbb{R}^{c_s\times h_s\times w_s}$ for images I^1, I^2 respectively, where $s \in \{1, 2, 3\}$, after the last three blocks in a ResNet50 [12] model.

Co-Attention Module: In order to predict changed regions in I_1, its feature maps must also embed information from I_2, and vice versa. We, therefore, wish to propagate the information embedded in f_s^1 and f_s^2 to each other in order to facilitate the computation of what has "changed". To permit this information exchange, we make use of the co-attention module [36]. Intuitively, each feature vector at location (x^1, y^1) in f_s^1 attends to feature vectors at all locations (x^2, y^2) in f_s^2 and is concatenated to their weighted sum (and vice-versa). This can be thought of as spatially warping the feature vectors of one image and concatenating with the other such that the two images are *registered*. Formally, we obtain the co-attended features $g_s^1 = [f_s^1 \parallel \psi(f_s^1, f_s^2)]$ and $g_s^2 = [f_s^2 \parallel \psi(f_s^2, f_s^1)]$, where $[\ \parallel\ ]$ is the concatenation operation (along channel c) and $\psi(\cdot)$ is the cross-attention mechanism defined as

$$\psi(f^q, f^k)_{cij} = \sum_l \sum_m A_{ijlm}.V_{clm} \tag{1}$$

where,

$$A_{ijlm} = \text{Softmax}(\sum_c Q_{cij}.K_{clm}, \text{dim=}l, m) \tag{2}$$

and,

$$Q = \mathbf{W}^q f^q,\ K = \mathbf{W}^k f^k,\ V = f^k \tag{3}$$

where $\mathbf{W}^q$ and $\mathbf{W}^k$ are learnable parameters. Thus, the feature maps g_s^1 and g_s^2 are conditioned on *both* the images and contain adequate information to localise the changes.

U-Net Decoder: Following this, we upsample and decode g_s^1, g_s^2 using the U-Net decoder (with skip connections from the encoder), modulated with scSE blocks [26], represented by $\Phi_D(\cdot)$ to produce feature maps h^1 and h^2 respectively, at the original image resolution.

Bbox Head: Finally, h^1 and h^2 are fed into a CenterNet head, which minimises the detection loss function as described in [39], to produce bounding boxes around changed regions in both the images.

[1]These datasets have not been released publicly at the time of writing.

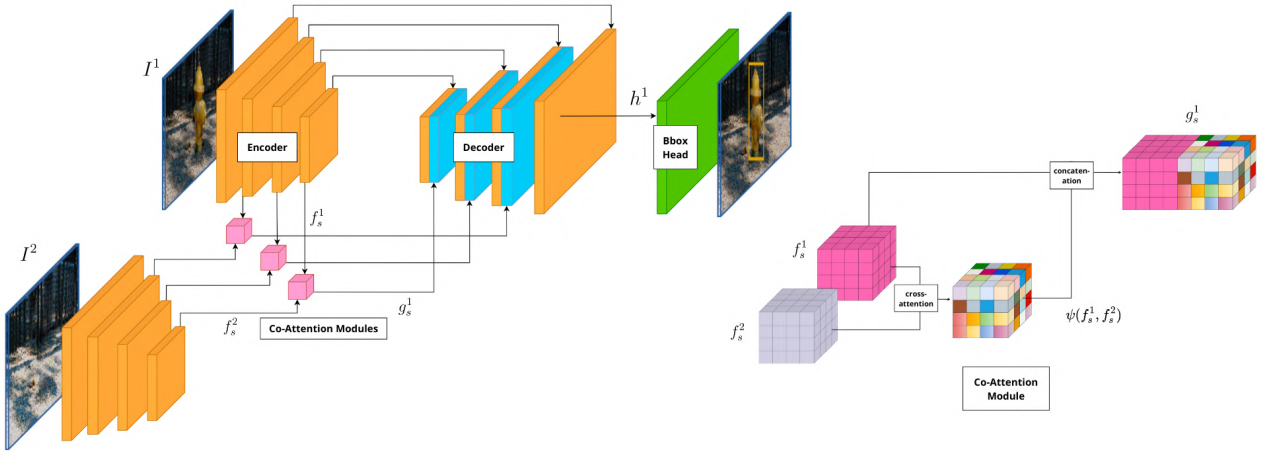


Figure 2. **Architecture:** Given two images I_1, I_2, an encoder produces feature maps f_s^1, f_s^2 respectively at multiple resolutions. A co-attention module is then used to compute conditioned feature maps g_s^1, g_s^2 that are impicitly registered with the other image. A U-Net style decoder is then applied to the original and conditioned features maps to produce feature maps h_1, h_2. Finally, the bbox detector head uses h_1, h_2 to produce bounding boxes for I_1, I_2 respectively. For brevity, we only show this pipeline for image I^1 (it is symmetric for image I^2). Please see Sec. 3 for details.

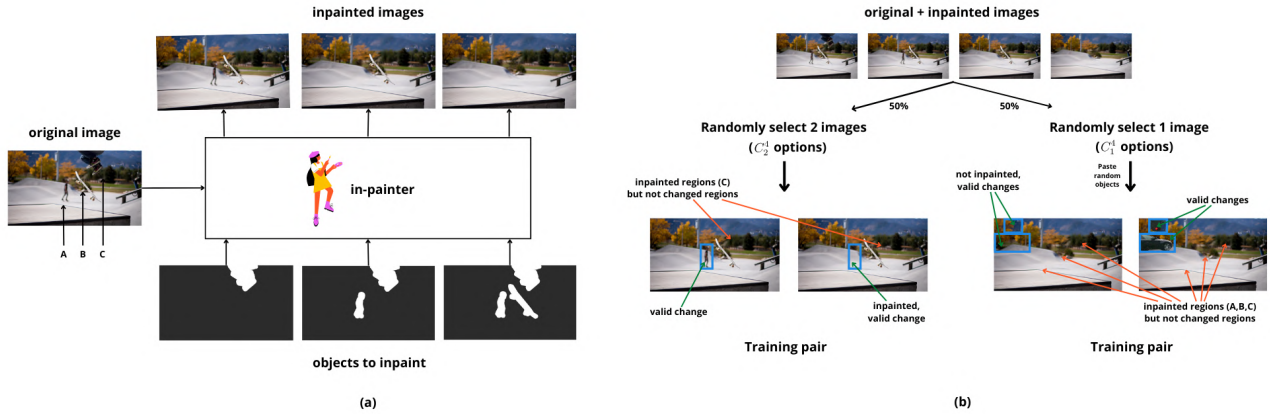


Figure 3. **Training data generation pipeline:** The figure above illustrates how we generate several image pairs with changed regions from a single COCO image. Given an original COCO image, we first (a) use an in-painting method to compute several images with inpainted regions. Then (b) given the original image along with inpainted images, we randomly sample an image pair for training, along with their ground truth bounding boxes, as shown. Notice that the image pairs can have inpainted regions that are not valid changes. This prevents the model from collapsing to simply learning inpainitng noise patterns. Please see Sec. 4 for details.

4. No Change Detection Dataset? No Problem

Much of the recent success of deep learning methods is attributed to the availability of large-scale training datasets with reliable annotations. Currently, however, there are no publicly available datasets for the change detection problem, as formulated in this work. To avoid curating and manually labelling thousands of image pairs with changes, we propose a procedure to leverage existing large-scale image datasets and state-of-the-art image inpainting methods to simulate visually realistic "changes". The complete training data processing pipeline is shown in Fig. 3 while we delineate the details below.

Inpainted changes: For this work, we make use of the COCO dataset [18], which comes with bounding boxes and segmentation masks for various objects in each image. Given a COCO image along with a binary segmentation mask of various objects in it, we inpaint the said objects using a state-of-the-art image inpainting method, LaMa

[32], to make the objects "disappear". The resulting inpainted image along with the original COCO image now constitute an image pair with changes (disappeared objects), for which we have the ground truth annotations (bounding boxes for the objects in the original COCO image).

Combating inpainting noise: Even though inpainting produces seemingly realistic changes, we noticed that the inpainted regions tend to have "noise" (as observed by other works in the literature [17, 37]). In order to discourage the model from simply learning this inpainting noise instead of learning the actual changes between images, we adopt the following two strategies:

- For each COCO image we obtain multiple inpainted images, each with a different subset of inpainted objects, out of which we randomly sample two. For instance, consider an image with 3 objects: A, B & C. Let's say we obtain two inpainted images: I_1 which only has object A (B & C have been inpainted), and I_2 which only has object B (A & C have been inpainted). In this case the model must predict two bounding boxes per image (B does not exist in I_1 and A does not exist in I_2, hence two changed regions per image). At the same time, the model must learn to ignore inpainting noise for C which has disappeared in both the images and is therefore not a valid change. Thus we force the model to learn actual visually-salient changes.
- Aside from inpainted changes, we also "paste" random objects into the image (taken from a different random COCO image) to simulate changes. While these inserted objects seem visually unrealistic, it requires the model to predict the "missing" object in the original image, which does not have any inpainting noise.

Training dataset: We randomly select 60000 images from the COCO train subset as our "original" images. For each original image, we use LaMa [32] to generate $n \in \{1, 2, 3\}$ images, each with a different subset of objects inpainted, as described in the pipeline above. We then randomly split these 60000 samples (each with C_2^{n+1} image pairs) into training and validation set consisting of 57000 and 3000 samples respectively. Each image is resized to 256×256 pixels (due to computational constraints), along with the appropriate scaling of its ground truth change bounding boxes. Given an image pair, we apply random affine transformations (scale $\in [0.8, 1.5]$, translation $\in [-0.2, 0.2]$ and rotation $\in [-\frac{\pi}{6}, \frac{\pi}{6}]$) to each image independently and adjust the ground truth bounding boxes appropriately. In addition, we apply random colour jittering to make our model invariant to photometric changes. We note that the change annotations are class agnostic in that they do not have access to the COCO class labels, rather the only classification is that something has changed at the scale of the bounding box. The validation set is strictly used to pick the best model (with the lowest loss) for evaluation and does not inform the training in any other way.

5. Experiments

Given two images, under some geometric transformation from one another, we aim to localise the changed regions while being invariant to photometric changes. This section describes the datasets we used to test our model and various implementation details, along with the results.

5.1. Evaluation datasets

To evaluate the performance of our model, we contribute four test datasets as described below. Please see Fig. 4 for example image pairs.

COCO-Inpainted: We curate an inpainting-based test set from the COCO test subset. We divide this test set into 3 categories based on the size (*small*, *medium* and *large*, as defined in [1]) of the inpainted objects. Using the same methodology as described in Sec. 4, we curate 1655 image pairs for *small*, 1747 image pairs for *medium* and 1006 image pairs for *large*, giving us a total of 4408 image pairs for this test set. Furthermore, we apply random affine transformations to the images along with colour jittering. Due to the affine transformations and cropping there will be some regions of the image that have no correspondence in the other image. Please see the first example pair in Fig. 4 for reference.

Synthtext-Change: We use the pipeline described in [10] to synthetically add random text to "background" images and generate 5000 image pairs with text-based changes in a manner that is consistent with their geometry. We do not augment the images any further i.e. the images have an identity geometric and photometric transformation. Note that in order to simplify quantitative evaluation, the generated texts are reasonably-spaced letters of varying sizes. This avoids having to deal with letter-level, word-level and paragraph-level predictions, where the model groups spatially-close small letters into a single bounding box but predicts a bounding box for each letter for bigger font-sizes.

VIRAT-STD: To detect outdoor scene changes, we select 1000 image pairs at random from the STD dataset [13]. These image pairs are originally taken from the VIRAT Video Dataset [20], which has bounding box annotations for several objects in each video frame. Since STD does not come with ground truth bounding box annotations for changes, we use a best-effort automated pipeline to obtain the ground truth (with a small percentage of them manually verified by human-in-the-loop). Since the camera is static,

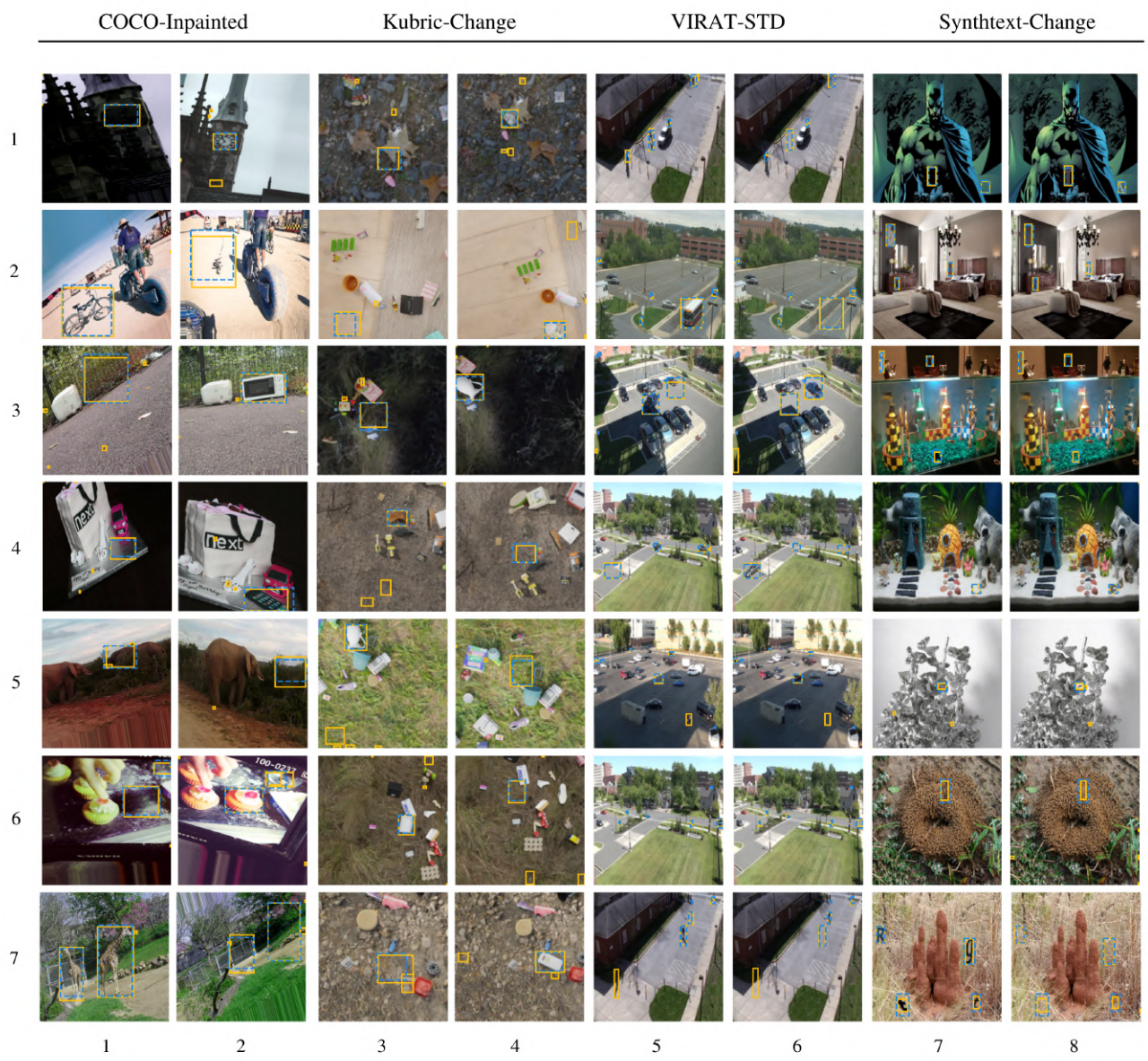


Figure 4. **Qualitative results:** We show the bounding box predictions (solid) of our model on all the test sets, along with the ground truth (dashed). Since the detection head outputs 100 bounding boxes per image (see Sec. 5.2), for the purpose of visualisation, we display the 5 most confident predictions. In case of multiple bounding boxes with significant overlap, we keep the most confident and suppress the others. Note the significant photometric changes in COCO-Inpainted, 3D geometric effects in Kubric-Change (notice the inside of the cup in row 2, col 3-4), detection of really small objects in VIRAT-STD (even picking up valid changes that are not part of the ground truth e.g. row 5, col 5-6) and very subtle letters in Synthtext-Change. We recommend that the reader zooms in on the individual image pairs for inspection.

there is an identity geometric transformation between the images (though there may be small motions of the camera due to wind etc.), but the photometric conditions may change due to time-of-day, weather conditions etc.

Kubric-Change: We use the recently introduced Kubric dataset generator [8] to curate 1605 realistic-looking image pairs with controlled changes. The scenes consist of a randomly selected set of 3D objects resting on a randomly textured ground plane. For a given scene, we iteratively remove objects from it and capture "before" and "after" image pairs. Unlike the datasets above where there is

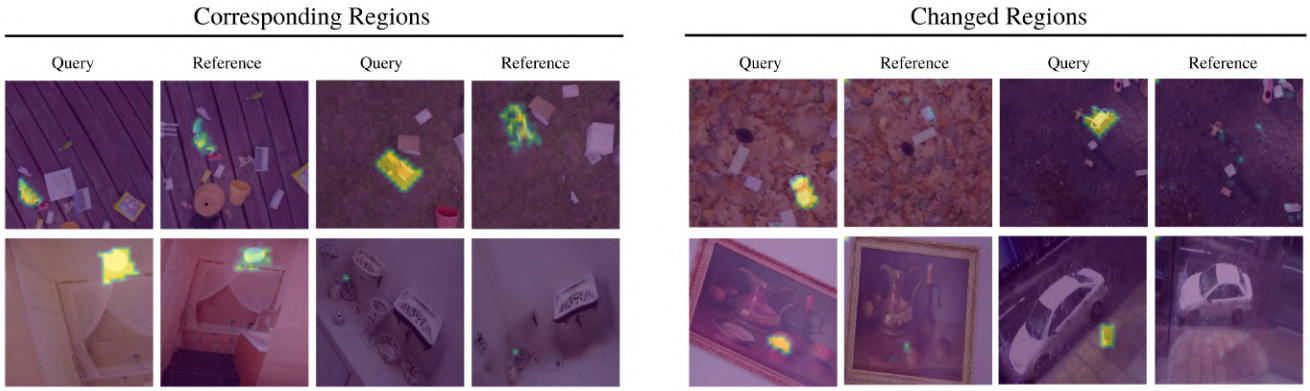


Figure 5. **Co-attention maps:** Given some pre-determined regions in I_1 (QUERY), we visualise the cross-attended regions in I_2 (REFERENCE) from the (spatially highest resolution) COAM layer of our model. The examples on the left show varying shape and size of QUERY regions, including the correspondence for a single pixel. The examples on the right show cases where the query region selected has no correspondence on the right (as the object is missing) – in these cases the attention map correctly does not highlight a region in the REFERENCE images. It is evident that the model has not only learnt to establish corresponding regions between the two images, but also fine-grained point-to-point correspondences.

backbone	# attn modules	attn type	scSE	geom transformation (train & test)	coco-inpainted test set (AP) small	medium	large	all
ResNet18	2	COAM	✗	affine	0.08	0.16	0.26	0.11
ResNet18	3	COAM	✗	affine	0.32	0.49	0.49	0.37
ResNet50	3	NOAM	✗	affine	0.15	0.32	0.49	0.21
ResNet50	3	COAM	✗	affine	**0.46**	0.74	0.70	0.58
ResNet50	3	COAM	✓	affine	**0.46**	**0.79**	**0.85**	**0.63**
ResNet50	3	COAM	✓	identity	0.60	0.89	0.94	0.73
ResNet50	3	NOAM	✓	identity	**0.68**	**0.93**	**0.95**	**0.79**

Table 1. **Ablation study:** We ablate various components of our model and report the AP on two variants (*affine*, *identity*) of the COCO-inpainted test set. Note that due to out-of-bounds cropping when applying geometric transformations, *affine* and *identity* test sets do not necessarily have the same number of changes, and methods trained and tested on one should not be directly compared with the other.

a planar geometric transformation between the images (affine or identity), for these image pairs the camera centre moves. Since the scene is 3D there can be parallax and occlusion/disocclusion changes between the image pairs.

5.2. Implementation

We use ResNet50 [12] as the encoder for our U-Net model (with ImageNet pre-trained weights), with 5 blocks (1-5), where we apply the co-attention module to the feature maps of blocks 3-5. The U-Net decoder also has 5 blocks with depths $(256, 256, 128, 128, 64)$, along with scSE blocks [26]. The CenterNet head is implemented as described in [39] with the hidden channel dimension of 64 and is configured to predict 100 bounding boxes per image. The overall model has 49.5M trainable parameters and is trained on 2 P40 GPUs for 200 epochs, using the DDP training strategy with a batch size of 16. We use Adam [14] to optimise the overall objective with learning rate of 0.0001 and weight decay of 0.0005.

5.3. Evaluation Metrics

To quantitatively evaluate our model, we compute the Average Precision (AP) metric defined in [5], as is standard. We emphasise the fact that for each image pair, the model outputs bounding boxes of changed regions for *both* the images and is evaluated as such.

5.4. Ablation

To study the effect of various modules of our method, we ablate different components of our model and show its performance on the 3 subsets (*small*, *medium* and *large*) of the COCO-Inpainted test set. As evident from Table 1, using more attention modules (3 instead of 2), using a bigger model (ResNet50 as opposed to ResNet18) and adding scSE blocks [26] all lead to an improvement in the results.

Furthermore, given two images under affine transformation, we recognise that it is possible to register them if their transformation matrix is known. Consequently, if we know a priori that the images are registered, we note that it is pos-

test set	COCO-Inpainted	Synthtext-Change	VIRAT-STD	Kubric-Change
type	inpainting	synthetic	real	photo-realistic sim
fixed camera	✓	✓	✓	✗
geometric transformation	Affine	None	None	3D
# image pairs	4408	5000	1000	1605
result (AP)	0.63	0.89	0.54	0.76

Table 2. **Quantitative results:** We report the AP of our model (ResNet50, 3 COAM layers, with scSE blocks) on various test sets.

sible to replace the co-attention module (COAM) with a simpler module, which we call no-attention module (NOAM), which simply concatenates the features maps from the two images i.e. $\psi(f^q, f^k) = f^k$ in eq. 1. The results from Table 1 show that COAM is almost on par with NOAM when the images are under an identity transformation, however, NOAM is much worse than COAM under geometric transformations.

5.5. Results

We take our model (ResNet50 backbone, 3 COAM layers, with scSE blocks) trained on the dataset described in Sec. 4 (with affine transformation) and evaluate it on 4 test sets without any further training/finetuning. We show some qualitative predictions by the model in Fig. 4 on each of the test sets and report the average precision values in Table 2. To the best of our knowledge, there are no existing works that tackle the change detection problem using a bounding-box based method which makes it difficult to compare our method with prior-art.

Our results show that not only is our method able to detect changes under extreme affine transformation and colour-jittering for the COCO-Inpainted test set, but also it is able to generalise zero-shot to changed image pairs procured from very different data distributions. Particularly, we note that even though our model is only trained using affine transformations, it produces impressive results on the Kubric-Change test set, where the changed image pairs are no longer related by a homography due to the movement of camera centre and the fact that the objects in the scene are 3D.

In Fig. 5 we show the visualisation of attention maps from the co-attention module. Specifically, given feature maps $f_s^1 \in \mathbb{R}^{C\times I\times J}$ and $f_s^2 \in \mathbb{R}^{C\times L\times M}$ of images I_1 and I_2 respectively, we obtain $A \in \mathbb{R}^{I\times J\times L\times M}$ using eq. 2. Then, for a set of query locations q in f_s^1, the co-attention map G, given by

$$G_{lm} = \max_{(i,j)\in q} A_{ijlm}, \tag{4}$$

represents the attended locations in f_s^2. It is evident from the visualisations that the model has learnt to establish correspondences between the two images, which is a logical step towards finding the changes.

6. Conclusion

Humans have a hard time finding changes in a scene – which is why we tend to find "spot the difference" tasks to be quite challenging. Adding viewpoint and photometric changes on top of this already difficult problem further elevates its perplexity. In this work we tackled the problem of automatically detecting changes in two images of the same scene under some geometric transformation, while ignoring nuisance factors such as photometric changes. We study a new formulation of this problem and treat it as a bounding-box based detection problem. Due to the lack of a large-scale training dataset for this problem, we proposed a training data generation pipeline that leverages existing datasets (or any arbitrary collection of images for that matter) and off-the-shelf image inpainting methods. Finally, we proposed and trained a novel neural network (in an end-to-end manner using a standard detection loss [39]) and showed that it is able to successfully zero-shot detect changes (without any finetuning or sim2real training) on several new benchmarks.

Limitations: The method proposed in this work largely focuses on the detection of changes in *things* rather than *stuff* (as defined in [4]). While it is likely that the trained model has the capacity to detect *stuff* changes, we have not investigated this. Furthermore, since the trained model is a *change* detector and not an *object* detector, it may group several overlapping changed objects into a single bounding box (as a single changed "object"). Finally, due to the nature of the training data, the model was largely tested on relatively planar scenes with mild occlusions/dis-occlusions. Going beyond, to general two-view scenes, with significant changes in the camera pose and challenges such as parallax and severe occlusions/dis-occlusions is a natural direction for future work.

Acknowledgements: We would like to thank Charig Yang, Laurynas Karazija, Luke Melas-Kyriazi, Aleksandarv (Suny) Shtedritski and Yash Bhalgat for proof-reading the paper. This research is supported by EPSRC Programme Grant VisualAI EP/T028572/1 and a Royal Society Research Professorship RP\R1\191132.

References

[1] COCO Detection Evaluation, https://cocodataset.org/#detection-eval.

[2] Pablo F. Alcantarilla, Simon Stent, German Ros, Roberto Arroyo, and Riccardo Gherardi. Street-view change detection with deconvolutional networks. In Proceedings of Robotics: Science and Systems, AnnArbor, Michigan, June 2016.

[3] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. In ACM Transactions on Graphics (ToG), volume 28, page 24. ACM, 2009.

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1209–1218, 2018.

[5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2):303–338, 2010.

[6] Mohammed E Fathy, Quoc-Huy Tran, M Zeeshan Zia, Paul Vernaza, and Manmohan Chandraker. Hierarchical metric learning and matching for 2D and 3D geometric correspondences. In Proc. ECCV, 2018.

[7] Aito Fujita, Ken Sakurada, Tomoyuki Imaizumi, Riho Ito, Shuhei Hikosaka, and Ryosuke Nakamura. Damage detection from aerial images via convolutional neural networks. In 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), pages 5–8, 2017.

[8] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3749–3761, 2022.

[9] Lionel Gueguen and Raffay Hamid. Large-scale damage detection using satellite imagery. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1321–1328, 2015.

[10] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[11] Ryuhei Hamaguchi, Shun Iwase, Rio Yokota, Yutaka Matsuo, Ken Sakurada, et al. Epipolar-guided deep object matching for scene change detection. arXiv preprint arXiv:2007.15540, 2020.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[13] Harsh Jhamtani and Taylor Berg-Kirkpatrick. Learning to describe differences between pairs of similar images. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[15] Zihang Lai, Erika Lu, and Weidi Xie. MAST: A memory-augmented self-supervised tracker. In Proc. CVPR, 2020.

[16] Yinjie Lei, Duo Peng, Pingping Zhang, Qiuhong Ke, and Haifeng Li. Hierarchical paired channel fusion network for street scene change detection. IEEE Transactions on Image Processing, 30:55–67, 2021.

[17] Ang Li, Qiuhong Ke, Xingjun Ma, Haiqin Weng, Zhiyuan Zong, Feng Xue, and Rui Zhang. Noise doesn't lie: Towards universal detection of deep inpainting. In Zhi-Hua Zhou, editor, Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pages 786–792. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Main Track.

[18] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In ECCV, 2014.

[19] Cristina Mata, Nick Locascio, Mohammed Azeem Sheikh, Kenny Kihara, and Dan Fischetti. Standardsim: A synthetic dataset for retail environments. In International Conference on Image Analysis and Processing, pages 65–76. Springer, 2022.

[20] Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, JK Aggarwal, Hyungtae Lee, Larry Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In CVPR 2011, pages 3153–3160. IEEE, 2011.

[21] Ariyo Oluwasanmi, Enoch Frimpong, Muhammad Umar Aftab, Edward Y. Baagyere, Zhiguang Qin, and Kifayat Ullah. Fully convolutional captionnet: Siamese difference captioning attention model. IEEE Access, 7:175929–175939, 2019.

[22] Dong Huk Park, Trevor Darrell, and Anna Rohrbach. Robust change captioning. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pages 4623–4632, 2019.

[23] Jin-Man Park, Jae-Hyuk Jang, Sahng-Min Yoo, Sun-Kyung Lee, Ue-Hwan Kim, and Jong-Hwan Kim. Changesim: Towards end-to-end online scene change detection in industrial indoor environments. In 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 8578–8585. IEEE, 2021.

[24] Julia Patriarche and Bradley Erickson. A review of the automated detection of change in serial imaging studies of the brain. J. Digit. Imaging, 17(3):158–174, Sept. 2004.

[25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[26] Abhijit Guha Roy, Nassir Navab, and Christian Wachinger. Concurrent spatial and channel 'squeeze & excitation'in fully convolutional networks. In International conference on medical image computing and computer-assisted intervention, pages 421–429. Springer, 2018.

[27] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. pages 61.1–61.12, 01 2015.

[28] Ken Sakurada, Mikiya Shibuya, and Weimin Wang. Weakly supervised silhouette-based semantic scene change detection. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 6861–6867, 2020.

[29] Nikolay Savinov, Lubor Ladicky, and Marc Pollefeys. Matching neural paths: transfer from recognition to correspondence search. In NeurIPS, 2017.

[30] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In Proc. ECCV, 2016.

[31] Simon Stent, Riccardo Gherardi, Björn Stenger, and Roberto Cipolla. Detecting change for multi-view, long-term surface inspection. In Xianghua Xie, Mark W. Jones, and Gary K. L. Tam, editors, Proceedings of the British Machine Vision Conference (BMVC), pages 127.1–127.12. BMVA Press, September 2015.

[32] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 2149–2159, 2022.

[33] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In Proc. ECCV, 2018.

[34] Qianqian Wang, Xiaowei Zhou, Bharath Hariharan, and Noah Snavely. Learning feature descriptors using camera pose supervision. In Proc. ECCV, 2020.

[35] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In Proc. CVPR, 2019.

[36] Olivia Wiles, Sébastien Ehrhardt, and Andrew Zisserman. Co-attention for conditioned image matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 15920–15929, 2021.

[37] Haiwei Wu, Jiantao Zhou, Jinyu Tian, and Jun Liu. Robust image forgery detection over online social network shared images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 13440–13449, 2022.

[38] Junhui Wu, Yun Ye, Yu Chen, and Zhi Weng. Spot the difference by object detection, 2018.

[39] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. arXiv preprint arXiv:1904.07850, 2019.