Structured Visual Search via Composition-aware Learning

Mert Kilickaya, Arnold W.M. Smeulders
QUvA Lab, University of Amsterdam
kilickayamert@gmail.com, a.w.m.smeulders@uva.nl

Abstract

This paper studies visual search using structured queries. The structure is in the form of a 2D composition that encodes the position and the category of the objects. The transformation of the position and the category of the objects leads to a continuous-valued relationship between visual compositions, which carries highly beneficial information, although not leveraged by previous techniques. To that end, in this work, our goal is to leverage these continuous relationships by using the notion of symmetry in equivariance. Our model output is trained to change symmetrically with respect to the input transformations, leading to a sensitive feature space. Doing so leads to a highly efficient search technique, as our approach learns from fewer data using a smaller feature space. Experiments on two large-scale benchmarks of MS-COCO [29] and HICO-DET [4] demonstrates that our approach leads to a considerable gain in the performance against competing techniques.

1. Introduction

Visual image search is a core problem in computer vision, with many applications, such as organizing photo albums [44], online shopping [19], or even in robotics [3, 38]. Two popular means of searching for images are either text-to-image [6, 26] or image-to-image [41, 53]. While simple, text-based search could be limited in representing the *intent* of the users, especially for the spatial interactions of objects. Image-based search can represent the spatial interactions, however, an exemplar query may not be available at hand. Due to these limitations, in our work, we focus on a structured visual search problem of compositional visual search.

The composition is one of the key elements in photography [40]. It is the spatial arrangement of the objects within the image plane. Therefore, composition offers a natural way to interact with large image databases. For example, a big stock image company already offers tools for its users to find images from their databases by composing a query [1]. The users compose an abstract, 2D image query where they arrange the location and the category of the objects of interest, see Figure 1.

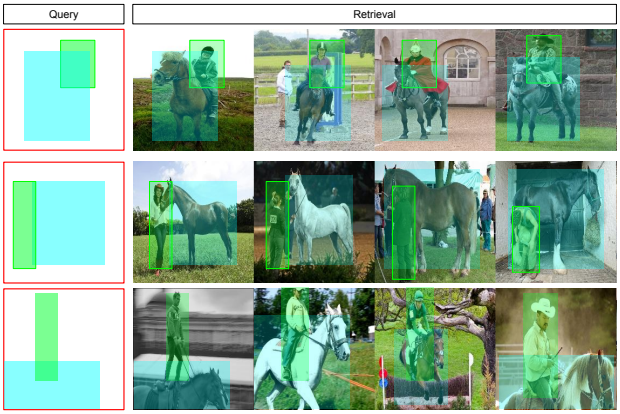

Figure 1: The compositional visual search takes a 2D canvas (left) as a query and then returns the relevant images that satisfy the object category and location constraints. Retrieval set (right) is in descending order by their mean Intersection-over-Union with the query canvas. Observe how small changes in the composition of the horse and the person lead to drastic transformations within the images. In this work, our goal is to learn these transformations for efficient compositional search.

Compositional visual search is initially tackled as a learning problem [51], recently using deep Convolutional Neural Networks (CNN) [31]. Mai *et al*. treats the problem as a visual feature synthesis task where they learn to map a given 2D query canvas to a 3 dimensional feature representation using binary metric learning which is then used for querying the database [31]. We identify the following limitations with this approach: *i)* The method requires a large-dimensional feature ($7 \times 7 \times 2048 \approx 100k$) to account for the positional and categorical information of the input objects, limiting the memory efficiency especially while searching across large databases. *ii)* The method requires a large-scale dataset ($\approx 70k$ images) for training, limiting the sample efficiency. *iii)* The method only considers bi-

nary relations between images, limiting the compositional-awareness. To overcome these limitations, in our work, we introduce composition-aware learning.

Compositional queries exhibit continuous-valued similarities between each other. Objects within the queries transform in two major ways: 1) Their positions change (translational transformation), 2) Their categories change (semantic transformation), see Figure 1. Our composition-aware learning approach takes advantage of such transformations using the principle of equivariance, see Figure 2. Our formulation imposes the transformations within the input (query) space to have a symmetrical effect within the output (feature) space. To that end, we develop novel representations of the input and the output transformations, as well as a novel loss function to learn these transformations within a continuous range.

Our contributions are three-fold:

I. We introduce the concept of composition-aware learning for structured image search.

II. We illustrate that our approach is efficient both in feature-space and data-space.

III. We benchmark our approach on two large-scale datasets of MS-COCO [29] and HICO-DET [4] against competitive techniques, showing considerable improvement.

2. Related Work

Compositional Visual Search. Visual search mostly focused on text-to-image [5, 6, 26, 33, 45, 47] or image-to-image [2,12,13,18,25,27,41,42,43,46,53] search. Text-to-image is limited in representing the user intent, and a visual query may not be available for image-to-image search. Recent variants also combine the compositional query either with text [11] or image [30]. In this paper, we focus on compositional visual search [31, 37, 51]. A user composes an abstract, 2D query representing the objects, their categories, and relative locations which is then used to search over a potentially large database. A successful example is VisSynt [31] where the authors treat the task as a visual feature synthesis problem using a triplet loss function. Such formulation is limited in the following ways: 1) VisSynt is high dimensional in feature-space ($100k$ dimensional), limiting memory efficiency, 2) VisSynt requires a large training set ($70k$ examples), limiting data efficiency, 3) VisSynt does not consider the compositional transformation between queries due to binary nature of the triplet loss [15], limiting the generalization capability of the method. In our work, inspired by the equivariance principle, we propose composition-aware learning to overcome these limitations and test our efficiency and accuracy on two well-established benchmarks of MS-COCO [29] and HICO-DET [4].

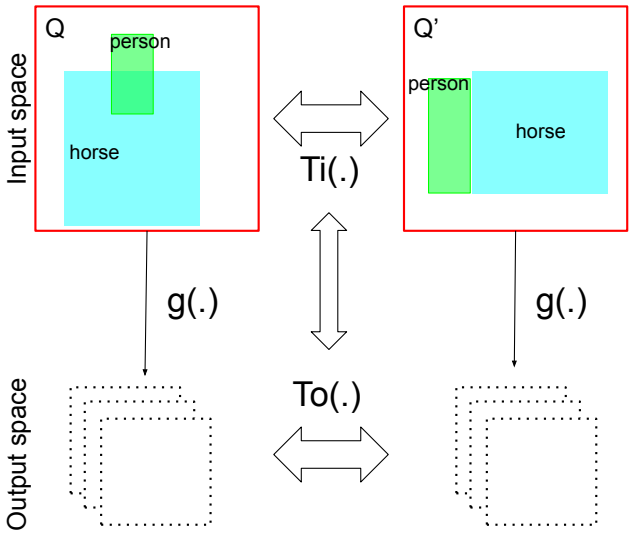

Figure 2: At the core of our technique is the principle of equivariance, which enforces a symmetrical change within the input and output spaces. We achieve this via mapping a query Q and its transformed version $Q' = T_i(Q)$ to a feature space where the transformation holds $g(Q') = T_o(g(Q))$.

Learning Equivariant Transformations. Equivariance is the principle of the symmetry: Any change within the input space leads to a symmetrical change within the output space. Such formulation is highly beneficial, especially for model and data efficiency [10]. In computer vision, equivariance is used to represent transformations such as object rotation [8, 48, 49], object translation [20, 32, 50, 52] or discrete motions [16, 17]. Our composition-aware learning approach is inspired by these works, as we align the continuous transformation between the input (query) and output (feature) spaces, see Figure 2.

Continuous Metric Learning. Continuous metric learning takes into account the continuous transformations between the image instances [23, 24, 35], since such relationships can not be modeled with conventional metric learning techniques [7, 15]. Recently, Kim *et al*. [23] proposed LogRatio, a loss function that matches the relative ratio of the input similarities with the output feature similarities. It yields significant gain over competing methods for pose and image caption search. Since compositional visual search is a continuous-valued problem, we bring LogRatio as a strong baseline to this problem. LogRatio intrinsically assumes a dense set of relevant images given an anchor point for an accurate estimation. However, compositional visual search follows Zipf distribution [36], where, given a query, only a few images are relevant, limiting LogRatio performance.

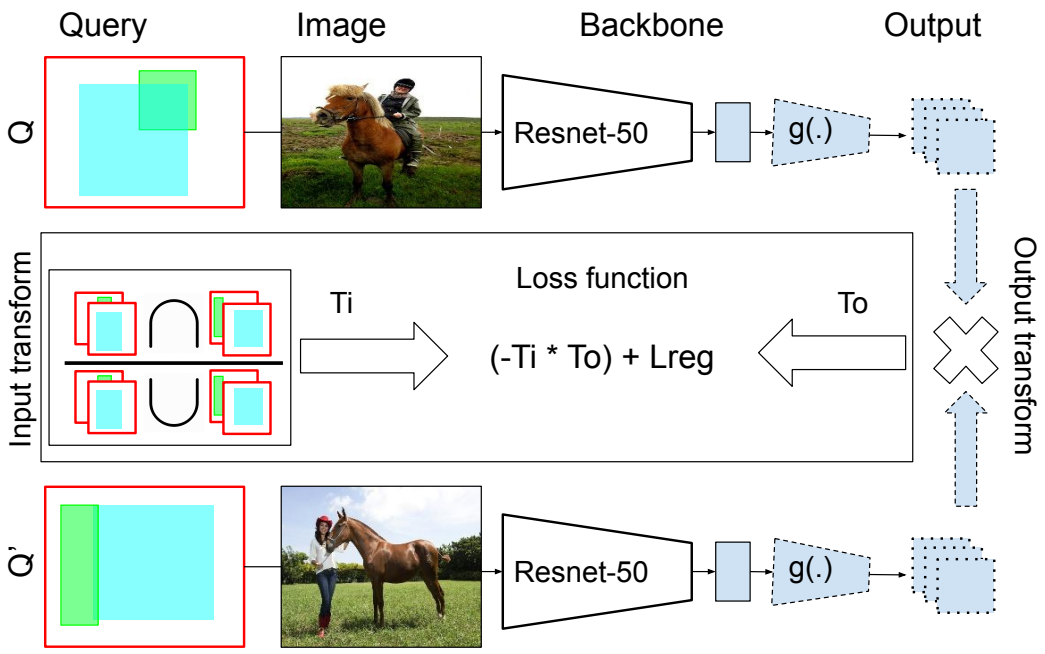

Figure 3: Our composition-aware learning approach. Our approach is trained with pairs of queries (Q, Q') with identical backbones. 1) Given a pair of queries, we sample the corresponding images and feed them through a frozen ResNet-50 up to layer-4 of size $7 \times 7 \times 2048$. 2) Then, we process these activations with our light-weight 3-layer CNN $g(\cdot)$ to map the channel dimension to a smaller size (*i.e.* $2048 \rightarrow 256$) while preserving the spatial dimension of 7. 3) In the mean-while, we compute the input (T_i) and the output (T_o) transformations, which are then forced to have similar values using the loss function.

3. Composition-aware Learning

Our method consists of three building blocks:

1. Composition-aware transformation that computes the transformations in the input and output space,
2. Composition-aware loss function that updates the network parameters according to the divergence of input-output transformations,
3. Composition-equivariant CNN, used as the backbone to learn the transformation.

Method Overview. An overview of our method is provided in Figure 3. Our method takes as an input a 2D compositional query $q \in \mathbb{R}^{H \times W}$, where H, W are the height and width of the query canvas. This query contains a set of objects, along with their categories and positions (in the form of bounding boxes). The goal of our method is, given a target dataset of images, we want to retrieve the top-k images that are most relevant to the query q – *i.e.* relevant to both the objects and their positions. Each image I can initially be represented as feature $x \in \mathbb{R}^{H' \times W' \times C'}$ using the last convolutional layer of an off-the-shelf, ImageNet pre-trained deep CNN, *e.g.* ResNet-50 [14]. Such feature x preserves the spatial information as well as the object category information within the image I. Furthermore, we assume access to a tuple (c, x, I), where $c \in \mathbb{R}^{H \times W \times C}$ is a compositional map constructed using the object categories and bounding boxes of the query q. In addition, let $q' = T(q)$ be the transformed version of the query q, and (c', x', I') are the corresponding composition map, CNN feature and the image. The transformation T can correspond to a translation of object location(s), or a change in object categories in q. Our method trains a 3-layer CNN $g_\Theta(\cdot)$ with the parameters Θ, by minimizing the following objective function:

$$\min_{\Theta}(L_{comp}(T_i(c, c'), T_o(g_\Theta(x), g_\Theta(x')))), \qquad (1)$$

where T_i measures the input transformation between compositional maps c and c', and T_o measures the transformation between output feature maps $g(x)$ and $g(x\prime)$, and L_{comp} is the composition-aware loss function measuring the discrepancy between these transformations. In the following, we first describe the compositional map c, and the input and the output transformations T_i and T_o. Then, we describe composition-aware loss function L_{comp}. Finally, we describe our CNN architecture $g_\Theta(\cdot)$ that learns the mapping. We drop Θ from now for the sake of clarity.

3.1. Composition-aware Transformation

The goal of the composition-aware transformation is to quantify the amount of transformation between the input compositions (c, c') and output feature maps $(g(x), g(x'))$ in the range $[0, 1]$. For this, first, we construct compositional maps from the input user queries, then we measure the input transformation using these maps, and finally we describe the output transformation.

Constructing compositional map c. First, given a user query q that reflects the category and the position of the objects, we create a one-hot binary feature map c of size $\mathbb{R}^{H,W,C}$ where $[H, W]$ are the spatial dimension of the composition map ($H = W = 32$), and C is the number of object categories (*i.e.* 80 for MS-COCO [29]). In this map, only the corresponding object locations and the categories are set to $1s$ and otherwise $0s$. This simple map encodes both the positional and categorical information of the input composition, which we will then use to measure the transformation within the input space. We apply the same procedure to the transformed query q' which yields c'. Now given the pair of compositional maps (c, c'), we can quantify the input transformation.

Input transformation T_i. Then, our goal is to measure the similarity between these two compositions as:

$$T_i(c, c') = \frac{\sum_{xyz}(c_{xyz} \cdot c'_{xyz})}{\sum_{xyz} \mathbb{1}(c_{xyz} + c'_{xyz})}, \tag{2}$$

where $\mathbb{1}$ is an indicator function that is 1 for only non-zero pixels. This simple expression captures the proportion of the intersection of the same-category object locations in the numerator and the union of the same-category object locations in the denominator. T_i output is in the range $[0, 1]$, and will return 1 if the two compositions c and c' are identical in terms of object location and the categories, and 0 if no objects share the same location. T_i will smoothly change with the translation of the input objects in the compositions. Given the input transformation, we now need to compute the output transformation which will then be correlated with the changes within the input space.

Output transformation T_o. Output transformation is computed as the dot product between the output features as follows:

$$T_o(g(x), g(x')) = g(x) \times (g(x'))^\top, \tag{3}$$

where $(g(x'))^\top$ is the transpose of the output feature $g(x')$. We choose the dot product due to its simplicity and convenience in a visual search setting. T_o can take arbitrary values in the range $[-\infty, \infty]$. In the following, we describe how to bound these values and measure the discepancy between the input-output transformations T_i and T_o.

3.2. Composition-aware Loss

Given the input-output transformations, we can now compute their discrepancy to update the parameters Θ of the network $g(\cdot)$. A *naive* way to implement this would be to minimize the Euclidean distance between the input-output transformations as:

$$\min_{\Theta} \|\mathbf{T_i} - \sigma(\mathbf{T_o})\|, \tag{4}$$

where $\sigma(\cdot)$ is the exponential non-linearity $\frac{1}{1+\exp(\cdot)}$ to bound T_o in range $[0, 1]$. However, such a function generates unbounded gradients therefore leading to instabilities during training [28], and reducing the performance, as we show through our experiments. Instead, cross entropy is a stable and widely used function that is used to update the network weights. However, cross entropy can only consider binary labels as $(0, 1)$ whereas in our case the transformation values vary within $[0, 1]$. To that end, we derive a new loss function inspired by the cross entropy that can still consider in-between values.

Consider that our goal is to maximize the correlation between input-output transformations as:

$$\max_{\Theta}(T_i \cdot \sigma(T_o^\top)). \tag{5}$$

We can also equivalently minimize the negative of this expression due to convenience:

$$\min_{\Theta}(-T_i \cdot \sigma(T_o^\top)). \tag{6}$$

The divergence of T_o and T_i at the beginning of the training leads to instabilities during the training. To overcome this, we include additional regularization via the following two terms as:

$$\min_{\Theta}(T_o - T_i \cdot T_o^\top + \log(1 + \exp(-T_o))), \tag{7}$$

where the two terms T_o and $\log(1+\exp(-T_o))$ penalize for larger values of T_o in the beginning of the training, leading to lesser divergence from T_i. To further avoid over-flow, the final form of the regularizer terms are:

$$\min_{\Theta}(\max(T_o, 0.) - T_i \cdot T_o^\top + \log(1 + \exp(-\|T_o\|))). \tag{8}$$

This is the final expression for L_{comp} which we use throughout the training of our network $g(\cdot)$.

3.3. Composition-Equivariant Backbone

Our model $g(\cdot)$ is a lightweight 3-layers CNN that maps the bottleneck representation x obtained from the pre-trained network ResNet-50 of dimension $\mathbb{R}^{7\times7\times2048}$ to a smaller channel dimension of the same spatial size, *i.e.* $\mathbb{R}^{h\times w\times C}$, such as $7 \times 7 \times 256$ unless otherwise stated. Our intermediate convolutions are $2048 \rightarrow 1024 \rightarrow 512 \rightarrow 256$. The first two convolutions use 3×3 kernels whereas the last layer uses 1×1. We use stride$= 1$ and apply zero-padding to preserve the spatial dimensions which are crucial for our task. We use $LeakyReLU$ with slope parameter $s = 0.2$, batch-norm and dropout with $p = 0.5$ in between layers. We do not apply any batch-norm, dropout, or $LeakyReLU$ at the output layer as this leads to inferior results.

Since our goal is to preserve positional and categorical information, a network with standard layers may not be a proper fit. Convolution and pooling operations in standard networks are shown to be lacking translation (shift) equivariance, contrary to wide belief [52]. To that end, we use the anti-aliasing trick suggested by [52] to preserve shift equivariance throughout our network. Specifically, before computing each convolution, we apply a Gaussian blur on top of the feature map. This simple operation helps to keep translation information within the network layers.

4. Experimental Setup

4.1. Datasets

Constructing Queries. To evaluate our method objectively, without relying on user queries and studies, we rely on large-scale benchmarks with bounding box annotations. We evaluate our method on MS-COCO [29] and HICO-Det [4]. The training is only conducted on MS-COCO. Given an image, we select at most 6 objects based on their area as is the best practice in [31].

MS-COCO. MS-COCO is a large-scale object detection benchmark. It exhibits 80 object categories such as animals (*i.e*. dog, cat, zebra, horse) or house-hold objects. The dataset contains $120k$ training and $5k$ validation images. We split the training set to two mutually exclusive random sets of $50k$ training and $70k$ gallery images. The number of objects in each image differs in the range $[1, 6]$.

HICO-DET. HICO-DET is a large-scale Human-object interaction detection benchmark [4, 21]. HICO-DET builds upon 80 MS-COCO object categories, and collects interactions for 117 different verbs, such as ride, hold, eat or jump, for 600 unique `<verb, noun>` combinations. Interactions exhibit fine-grained spatial configurations which makes it a challenging test for the compositional search. The dataset includes $37k$ training and $10k$ testing images. The training images are used as the gallery set and the testing set is used as the query set. A unique property of the dataset is that 150 interactions have less than 10 examples in the training set, which means a query can only match very few images within the gallery set, leading to a challenging visual search setup [22]. HICO-DET is only used for evaluation.

4.2. Evaluation Metrics

We evaluate the performance of the proposed model with three metrics. Standard mean Average Precision metric as is used in VisSynt [31]. Also, we borrow continuous Normalized Discounted Cumulative Gain (cNDCG) and mean Relevance (mREL) metrics used in continuous metric learning literature [23, 24, 35] All metric values are based on the mean Intersection-over-Union (mIOU) scores between a query and all gallery images described below. For all three metrics, higher indicates better performance.

4.2.1 Mean Intersection-over-Union

To measure the relevance between a query and a retrieved image, we resort to mean Intersection-over-Union as is the best practice [31]. Concretely, to measure the relevance between a Query q and a retrieved image r

$$mIOU(q, r) = \frac{1}{|B_q|} \sum_{b_i \in B_Q} \max_{b_j \in B_I} \mathbb{1}(k(b_i) = k(b_j)) \frac{b_i \cap b_j}{b_i \cup b_j}, \tag{9}$$

where B_Q and B_I represents all the available objects in the query Q and retrieved image I respectively, $\mathbb{1}$ is an indicator function that checks whether objects i and j are from the same class k, which is then multiplied with the intersection-over-union between these two regions. This way, the metric measures both the spatial and semantic localization of the query object.

4.2.2 Metrics

mAP. Based on the relevance score, we use mean Average Precision to measure the retrieval performance. We first use a heuristic relevance threshold ≥ 0.30 as recommended in [31], to convert continuous relevance values to discrete labels. Then, we measure the mAP values $@\{1, 10, 50\}$.

mAP metric does not respect the continuous nature of the compositional visual search since it binarizes continuous relevance values with a heuristic threshold. To that end, we resort to two additional metrics, continuous adaptation of NDCG and mean Relevance values which are used to evaluate continuous-valued metric learning techniques in [23, 24, 35].

cNDCG. We make use of the continuous adaptation of the Normalized Discounted Cumulative Gain as follows:

$$cNDCG(q) = \frac{1}{Z_k} \sum_{i=1}^{K} \frac{2^{r_i}}{\log_2(i+1)}, \quad (10)$$

that takes into account both the rank and the scores of the retrieved images and the ground truth relevance scores. In our experiments we report cNDCG@$\{1, 50, 100\}$.

mREL. mREL measures the mean of the relevance scores of the retrieved images per query, which is then averaged over all queries. In our experiments, we report mREL@$\{1, 5, 20\}$. We also note the **oracle** performance where we assume access to the ground truth mIOU values to illustrate the upper bound in the performance.

4.3. Performance Comparison

ResNet-50 [14]. We use the activations from layer-4 of ResNet-50 to retrieve images. In this work, we build upon this feature since it captures the object semantics and positions within the feature map of size $\mathbb{R}^{7\times7\times2048}$. We also experimented with the earlier layers, however we found that layer-4 performs the best. The network is pre-trained on ImageNet [9].

Textual. We assume access to the ground truth object labels for a query and retrieve images that contain the same set of objects. This acts as a textual query baseline and is blind to the spatial information.

VisSynt [31]. This baseline uses a triplet loss formulation coupled with a classification loss to perform a compositional visual search. We use the same backbone architecture $g(\cdot)$ and the same target feature ResNet-50 to train this baseline for a fair comparison.

LogRatio [23]. This method is the state-of-the-art technique in continuous metric learning, originally evaluated on human pose and image caption retrieval. In this work, we bring this technique as a strong baseline since the visual composition space also exhibits continuous relationships. We use the authors code [1] and the recommended setup. We convert mIOU scores to distance values as $1 - mIOU$ since the method minimizes the distances.

Implementation details. We use PyTorch [39] to implement our method. We use the same backbone ($g(\cdot)$) and the input feature (ResNet-50) for all the baselines. All the models are trained for 20 epochs using SGD with momentum ($= 0.9$). We use an initial learning rate of 10^{-2} which is decayed exponentially with 0.004 at every epoch. We use weight decay ($wd = 0.005$) for regularization. In practice, we compute input-output transformations between all examples within the batch to get the best out of each batch. We set the batch size to 36, and given each query in the batch, we sample 1 highly relevant and 1 less relevant examples for each query, which leads to an effective batch size of $36 \times 3 = 108$.

[1] https://github.com/tjddus9597/Beyond-Binary-Supervision-CVPR19

5. Experiments

In this Section, we present our experiments. For Experiments $1-2$, we use all three metrics $@k = 1$. For the third experiment of the State-of-the-Art comparison, we provide performance at different k values.

5.1. Ablation of Composition-aware Learning

Euclidean vs. Composition-aware loss. In our first ablation study, we compare the Euclidean loss described in Equation 4 with our composition-aware loss. The results are presented in Table 1.

	mAP	cNDCG	mREL
Euclidean	66.87	39.73	28.49
CAL (ours)	81.17	51.18	35.96

Table 1: Euclidean vs. Composition-aware loss.

It is observed that Composition-aware loss outperforms Euclidean alternative by a large-margin, confirming the effectiveness of the proposed loss function.

	mAP	cNDCG	mREL
Lingual	65.14	27.77	19.56
Visual (ours)	81.17	51.18	35.96

Table 2: Lingual vs. Visual input transformation.

Lingual vs. Visual transformation. In our second ablation study, we test the domain of the input transformation (Eq 2). In our work, we proposed a visual-based input transformation whereas VisSynt [31] utilizes a lingual-based input transformation using semantic Word2vec embeddings [34]. As can be seen from Table 2, vision-based transformation outperforms the lingual counterpart, since it can better encode the relationships within the visual world.

5.2. Feature and Data Efficiency

In this experiment, we test the efficiency. Specifically, we first test the feature-space efficiency to see how the performance changes with varying sizes of the query embedding. Second, we test the data-space efficiency by subsampling the training data.

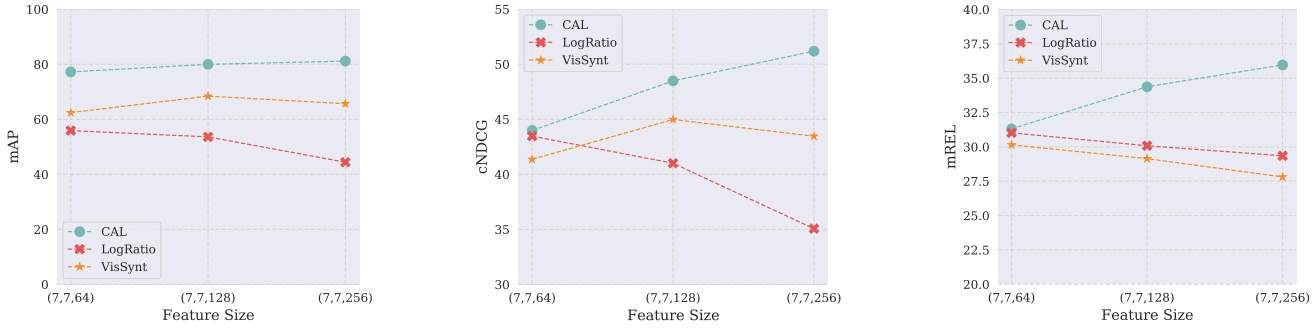

Figure 4: Feature efficiency. Our model performs better even when the feature-space is compact.

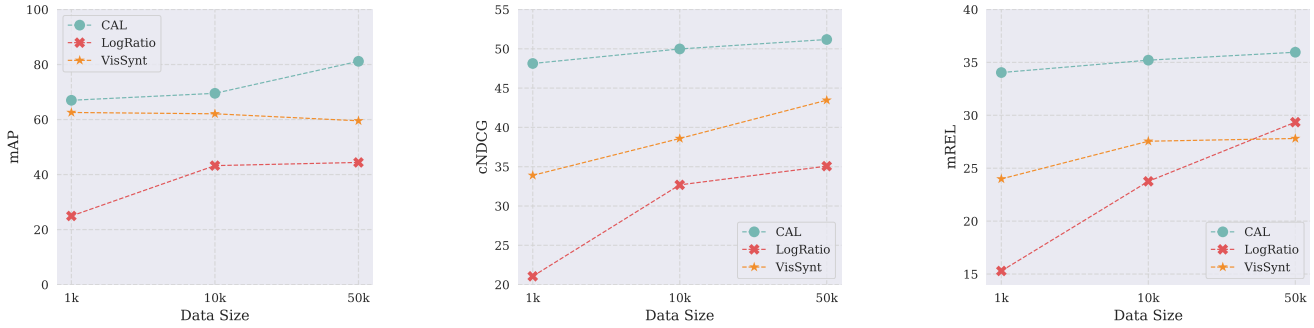

Figure 5: Data efficiency. Our model outperforms VisSynt and LogRatio within small data regime.

Feature-space efficiency. We change the feature embedding size by varying the number of channels as $64, 128, 256$ by keeping the spatial dimension of 7×7. We compare our approach to VisSynt [31] and LogRatio [23]. The results can be seen from Figure 4.

As can be seen, our approach performs the best for all metrics and across all feature sizes. This indicates that composition-aware learning is effective even when the feature size is compact (*i.e.* $7 \times 7 \times 64$). Another observation is that the performance of CAL increases with the increased feature size, whereas the performance of the two other techniques is lower. This indicates that CAL can leverage bigger feature sizes while other objectives tend to over-fit.

It is concluded that CAL is a feature-efficient approach for compositional visual search.

Data-space Efficiency. In this experiment we vary the number of training data as $1k, 10k, 50k$. The results can be seen from Figure 5.

Our method performs the best regardless of the training size. The gap in the performance is even more significant when the training set size is highly limited (*i.e.* $1k$ only), confirming the data efficiency of the proposed approach.

It is concluded that CAL can learn more from fewer examples by leveraging the continuous-valued transformations and the regularized loss function.

5.3. Comparison with the State-of-the-Art

In the last experiment, we compare our approach to competing techniques on MS-COCO in Figure 6 and HICO-DET in Figure 7 datasets.

As can be seen, our method outperforms the compared baselines in both datasets, and in 3 metrics. This confirms the effectiveness of composition-aware learning for object (MS-COCO) and object-interaction (HICO-DET) search. The results in HICO-DET are much lower compared to MS-COCO since 1) HICO-DET has a higher number of query images ($10k$ vs. $5k$), 2) Many queries have only a few relevant images within the gallery set (as can be seen from the oracle performance of only 0.19 mREL in Figure 7), 3) No training is conducted on HICO-DET, revealing the transfer-learning abilities of the evaluated techniques.

Qualitative analysis. Lastly, we showcase a few qualitative examples in Figure 8. First, as a sanity check, we illustrate single object queries (stop signs). As can be seen, our method successfully retrieves images relevant to the query category and the position. Then, we illustrate some object-interaction examples, such as human-on-bench, or human-with-tennis racket, or human-on-skateboard. Our model can still generalize to such examples, meaning that compositional learning benefits the case of the object interaction.

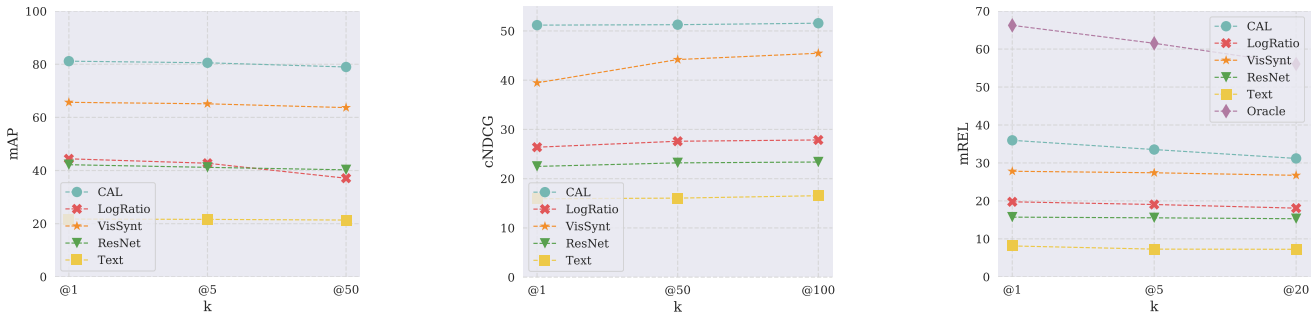

Figure 6: Benchmarking on MS-COCO [29]. Our method outperforms existing techniques for all three metrics.

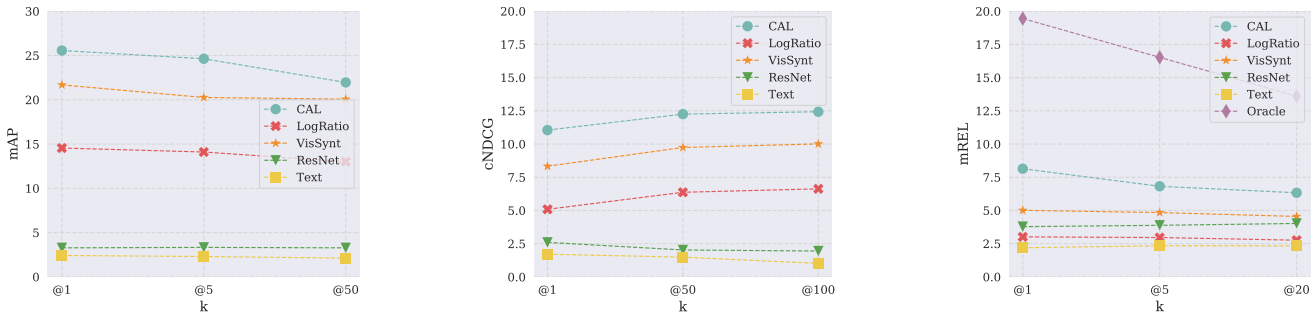

Figure 7: Benchmarking on HICO-DET [4]. Our method transfers better to HICO-DET dataset for object-interaction search.

Figure 8: Qualitative examples. First two rows show a single-object query, and last three rows show multi-object queries. As can be seen, our approach considers the object category, location and interaction into account while retrieving examples.

We illustrate a failure case in the last row, where our model retrieves a mix of snowboard-skateboard objects given the query of a skateboard. This indicates that our model performance can be improved by incorporating scene context, which we leave as future work.

6. Conclusion

In this work we tackled a structured visual search problem called compositional visual search. Our approach is based on the observation that the visual compositions are continuous-valued transformations of each other, carrying rich information. Such transformations mainly consists of the positional and categorical changes within the queries. To leverage this information, we proposed composition-aware learning which consists of the representation of the input-output transformations as well as a new loss function to learn these transformations. Our experiments reveal that defining the transformations within the visual domain is more useful than the lingual counterpart. Also, a regularized loss function is necessary to learn such transformations. Leveraging transformations with this loss function leads to an increase in the feature and data efficiency, and outperforms existing techniques on MS-COCO and HICO-DET. We hope that our work will inspire further research to incporporate structure for the structured visual search problems.

References

[1] Shutterstock compositional visual search. In *https://www.shutterstock.com/blog/composition-aware-search-tool*, 2017.

[2] Artem Babenko and Victor Lempitsky. Aggregating local deep features for image retrieval. In *ICCV*, 2015.

[3] Nils Bore, Rares Ambrus, Patric Jensfelt, and John Folkesson. Efficient retrieval of arbitrary objects from long-term robot observations. *RAS*, 2017.

[4] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018.

[5] Ding-Jie Chen, Songhao Jia, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. See-through-text grouping for referring image segmentation. In *ICCV*, 2019.

[6] Hui Chen, Guiguang Ding, Zijia Lin, Sicheng Zhao, and Jungong Han. Cross-modal image-text retrieval with semantic consistency. In *ACM MM*, 2019.

[7] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

[8] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *ICLR*, 2016.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[10] Carlos Esteves, Ameesh Makadia, and Kostas Daniilidis. Spin weighted spherical cnns. *arXiv preprint arXiv:2006.10731*, 2020.

[11] Ryosuke Furuta, Naoto Inoue, and Toshihiko Yamasaki. Efficient and interactive spatial-semantic image retrieval. *MTA*, 2019.

[12] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016.

[13] Yinzheng Gu, Chuanpeng Li, and Yu-Gang Jiang. Towards optimal cnn descriptors for large-scale image retrieval. In *ACM MM*, 2019.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[15] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[16] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *ICCV*, 2015.

[17] Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *CVPR*, 2016.

[18] Albert Jimenez, Jose M Alvarez, and Xavier Giro-i Nieto. Class-weighted convolutional features for visual instance search. *arXiv preprint arXiv:1707.02581*, 2017.

[19] Yushi Jing, David Liu, Dmitry Kislyuk, Andrew Zhai, Jiajing Xu, Jeff Donahue, and Sarah Tavel. Visual search at pinterest. In *ACM SIGKDD*, 2015.

[20] Osman Semih Kayhan and Jan C van Gemert. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *CVPR*, 2020.

[21] Mert Kilickaya, Noureldien Hussein, Efstratios Gavves, and Arnold Smeulders. Self-selective context for interaction recognition. *arXiv preprint arXiv:2010.08750*, 2020.

[22] Mert Kilickaya and Arnold Smeulders. Diagnosing rarity in human-object interaction detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 904–905, 2020.

[23] Sungyeon Kim, Minkyo Seo, Ivan Laptev, Minsu Cho, and Suha Kwak. Deep metric learning beyond binary supervision. In *CVPR*, 2019.

[24] Suha Kwak, Minsu Cho, and Ivan Laptev. Thin-slicing for pose: Learning to understand pose without explicit pose estimation. In *CVPR*, 2016.

[25] Herwig Lejsek, Björn ór Jónsson, Laurent Amsaleg, and Fririk Heiar Ásmundsson. Dynamicity and durability in scalable visual instance search. *arXiv preprint*, 2018.

[26] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Visual semantic reasoning for image-text matching. In *ICCV*, 2019.

[27] Yang Li, Zhuang Miao, Jiabao Wang, and Yafei Zhang. Nonlinear embedding neural codes for visual instance retrieval. *Neurocomputing*, 2018.

[28] Shuai Liao, Efstratios Gavves, and Cees GM Snoek. Spherical regression: Learning viewpoints, surface normals and 3d rotations on n-spheres. In *CVPR*, 2019.

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

[30] Jin Ma, Shanmin Pang, Bo Yang, Jihua Zhu, and Yaochen Li. Spatial-content image search in complex scenes. In *WACV*, 2020.

[31] Long Mai, Hailin Jin, Zhe Lin, Chen Fang, Jonathan Brandt, and Feng Liu. Spatial-semantic image search by visual feature synthesis. In *CVPR*, 2017.

[32] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. In *ICCV*, 2017.

[33] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.

[34] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint*, 2013.

[35] Greg Mori, Caroline Pantofaru, Nisarg Kothari, Thomas Leung, George Toderici, Alexander Toshev, and Weilong Yang. Pose embeddings: A deep architecture for learning to match human poses. *arXiv preprint*, 2015.

[36] Mark EJ Newman. Power laws, pareto distributions and zipf's law. *Contemporary physics*, 2005.

[37] Danilo Nunes, Leonardo Anjoletto Ferreira, Paulo E Santos, and Adam Pease. Representation and retrieval of images by means of spatial relations between objects. In *AAAI*, 2019.

[38] Hae Won Park and Ayanna M Howard. Retrieving experience: Interactive instance-based learning methods for building robot companions. In *ICRA*, 2015.

[39] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[40] Bryan Peterson. *Learning to see creatively: Design, color, and composition in photography*. Amphoto Books, 2015.

[41] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In *ECCV*, 2016.

[42] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *PAMI*, 2018.

[43] Ali S Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki. Visual instance retrieval with deep convolutional networks. *TMTA*, 2016.

[44] Kerry Rodden and Kenneth R Wood. How do people manage their digital photographs? In *SIGCHI*, 2003.

[45] Nikolaos Sarafianos, Xiang Xu, and Ioannis A Kakadiaris. Adversarial representation learning for text-to-image matching. In *ICCV*, 2019.

[46] Ran Tao, Arnold WM Smeulders, and Shih-Fu Chang. Attributes and categories for generic instance search from one example. In *CVPR*, 2015.

[47] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. Camp: Cross-modal adaptive message passing for text-image retrieval. In *ICCV*, 2019.

[48] Maurice Weiler and Gabriele Cesa. General e (2)-equivariant steerable cnns. In *NeurIPS*, 2019.

[49] Maurice Weiler, Fred A Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *CVPR*, 2018.

[50] Daniel Worrall and Gabriel Brostow. Cubenet: Equivariance to 3d rotation and translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 567–584, 2018.

[51] Hao Xu, Jingdong Wang, Xian-Sheng Hua, and Shipeng Li. Image search by concept map. In *SIGIR*, 2010.

[52] Richard Zhang. Making convolutional networks shift-invariant again. *ICML*, 2019.

[53] Liang Zheng, Yi Yang, and Qi Tian. Sift meets cnn: A decade survey of instance retrieval. *PAMI*, 2017.