Dual-Phase Framework for Cross-Domain Content Retrieval

Anirudh Kannan, B Sathish Babu*

Dept. of Computer Science and Engineering, RV College of Engineering, Bengaluru

Abstract

Cross-Domain content is nothing but the data from various source distributions. Traditional machine learning methods do not perform well at modelling a mapping function when the source and target domains are diverse, making the task of matching two data points from two richly varied distributions difficult. Although specific Deep Learning models can assist bridging this domain gap, they are computationally costly and involve extensive training. A robust Dual-Phase Framework is presented in this paper to perform Cross-Domain mapping with less processing resources than the existing approaches. The first phase of this framework embeds the data points in different domains into a common latent space, and the second phase maps these embeddings with each other. The two steps are detached and trained independently which requires lesser computational power to produce strong results. The proposed hypothesis is verified on a sketchy database, producing robust results comparable with existing baselines. The framework effectively demonstrates good generalization and performance with limited resources.

Keywords: *Cross-Domain, mapping function, domain gap, Dual Phase Framework, common latent space*

1.0. Introduction

Human beings are born with the ability to map out and compare disparate objects with a remarkable ability to generalise previous knowledge with new facts, allowing adaption and drawing quick conclusions. Hence, human beings can effortlessly recognise cross-domain relationships unlike Deep Learning Systems [1]. Emergence of deep learning has resulted in surge in data demand. During the deployment of these models in the production environment, however, the input features encountered may not be part of the same distribution as the training data, called the covariate shift in a domain. It results in inadequate performance translation from training to testing. Ability of a model with respect to generalisation is proportionate to its ability to manage covariate shifts.

A cross-domain problem occurs when two sets of data from independent sources have a domain gap. Deep neural networks exhibit low

* *Mail address: B Sathish Babu, Professor, Department of Computer Science and Engineering, RV College of Engineering, Bengaluru – 59*
Email: bsbabu@rvce.edu.in, Ph: 9844488329

generalisability to covariate domain shift. Comparing data from semantically diverse domains is difficult. Several deep learning models such as Autoencoders have been created to address the challenge of Cross-Domain Learning. These models demand huge data, and processing resources to train them, which is a significant drawback.

Training an Autoencoder on different pairs of pictures requires a huge network which must be trained end-to-end on numerous pairs of images. In general, there are $O(N^2)$ number of image pairings in the training data if there are N examples of each image. Because Autoencoders comprise multiple layers, training the model on the complete dataset takes long time and huge processing power (GPUs).

This paper presents a unique Dual Phase Framework for Cross-Domain Content Retrieval to achieve domain generalisation concerning any content modalities (Fig. 1). The latent space vectors are extracted from the input dataset in the first phase. The dimensionality of the features retrieved from both the distributions is the same. The second stage entails converging similar (semantically) latent vectors and diverging those that are different. Models of Phase-I and Phase-II are trained separately. Since the dimensionality is reduced after the first step, the computing resources required for training are lowered, resulting in a simple and smaller model utilized for the second phase. Making cross-domain pairings for training is also easier when lower-dimensional characteristics are used instead of the content itself. Although the number of pairings remains the same, the dimension of the data is decreased, resulting in decreased GPU resource consumption. Sketch-Based Image Retrieval task [2] was used to elucidate the usefulness of this approach. Sketch-Based Image Retrieval entails selecting most relevant images from a given set of images based on a query – a hand-drawn sketch. In this scenario, the two domains are single-channel drawings and multi-channel pictures.

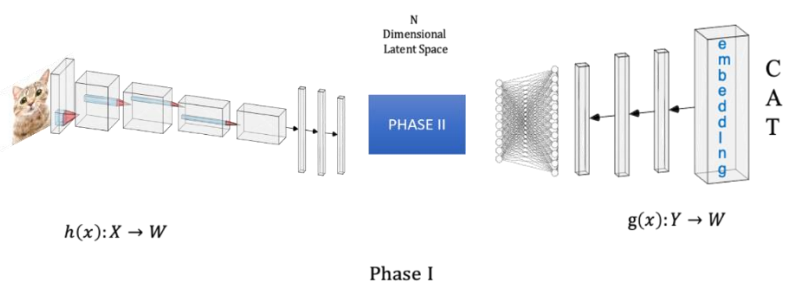


Fig. 1. Proposed Framework

Because these domains are part of different distributions, cross-domain matching is necessary to establish a qualitative relationship between the images [3]. The robustness of our proposal measured in terms of performance on the baselines using mean average precision (MAP) as a metric.

Research work on the use of Content-Based Image Retrieval systems in trademark search and verification [4], clip art generation [5, 6] and in documents [7] are available. Nevertheless, these works assumed that the sketches collected strokes and the geometric relationship between them for retrieval and matching [8, 9]. The authors concluded that these techniques are susceptible to geometric perspectives, occlusions, and noise.

Approaches based on edge map extraction were investigated, including the Canny edge detector and HOG [10]. These approaches could not effectively bridge the cross-domain gap. They follow the approach of combining source distributions into a single common distribution; however, this strategy was ineffective due to data loss. Authors [11, 12] investigated use of Siamese Networks, Autoencoders, and other deep learning models for feature extraction. These methods train the entire network at once and need many computing resources. The proposed framework extends previous approaches by offering an architecture that reduces the demand for computational resources and training time.

Current Cross-domain Comparison literature explores several techniques, including disentangling domain features [13] and structure-preserving learning [14]. Although these approaches are helpful, there is a shortage of literature exploring the training of complicated/large models. Studies examining training for robust performance in a conservative environment with minimal resources is scarce.

The architecture of Autoencoders is modified with the proposed framework. It includes detaching the feature extractor modules as separate, independent units as well as diminishing the dimension of the dataset and reducing the complexity of the cross-domain mapping need to be learned.

3.0. Framework Architecture

The two separate phases are discussed in depth. A data source is assumed to collect input vectors from a single distribution regarding the input to the pipeline.

3.1. Phase I

The first phase is the training of two mapping functions, which each turn the source data vectors into latent space vectors. Considering the two

sources of data, X_1 and X_2, as different distributions having a significant domain gap, two data points x_1 and x_2, belonging to distributions X_1 in m dimensional space and X_2 in n-dimensional space, are used as input vectors and W represents the latent space.

The latent space W consists of all the vectors spanned by the feature embeddings of the data points of X_1 and X_2. The two functions $h(x_1)$: $X_1 \rightarrow W$ and $g(x_2)$: $X_2 \rightarrow W$ are defined as:

$h(x_1) = w_1$ and

$g(x_2) = w_2$,

where w_1 and w_2 are the output vectors from this mapping.

Mapping functions h and g are trained on the pairs of data. If x1 and x2 input vectors are similar (measured by cosine similarity), then the output vectors are identical. These mapping functions integrate and embed the input vectors to produce clusters of quasi data points. Intrinsically similar source vectors belong to the same cluster.

From the point of view of deployment, it is possible to create a deep neural network to classify the data and use the penultimate network layer to embed the data in an N-dimensional space, where N is the number of neurons present penultimate layer. Another approach is to convert input feature vectors into a lower dimension output space using an unsupervised learning model. As explained above, this output space is the latent space W. Comparable vectors would therefore have similar latent vectors. Autoencoders may also be employed if the Autoencoder's latent space embedding may be used directly as phase I outputs.

3.2 Phase II

The Dual Phased Deep Learning Framework's second phase represents a function that effectively converges pairs of vector embeddings, producing the probability of a pair of vectors being semantically similar to the output. An Autoencoder Network and an appropriate distance metric, such as Euclidean distance, are often used in this second phase. To force the network to learn to distinguish between feature embeddings, either contrastive and triplet losses can be utilized. In reality, this entire pipeline can be considered an Autoencoder where the initial layers of the encoder and decoder are trained individually and frozen. The demand for GPU memory is lesser since the h(x) and g(x) functions are trained separately and not simultaneously. In addition to this, there is a reduction in time spent to train the entire pipeline.

4.0. Methodology

For validating the hypothesis proposed by the Dual-Phase Framework, Sketch-Based Image Retrieval as a reference task (Fig. 2). Given a hand-drawn sketch, this task aims to find the most relevant pictures from a set of photographs. The semantic closeness between the query sketch and the returned picture referred to it as relevance.

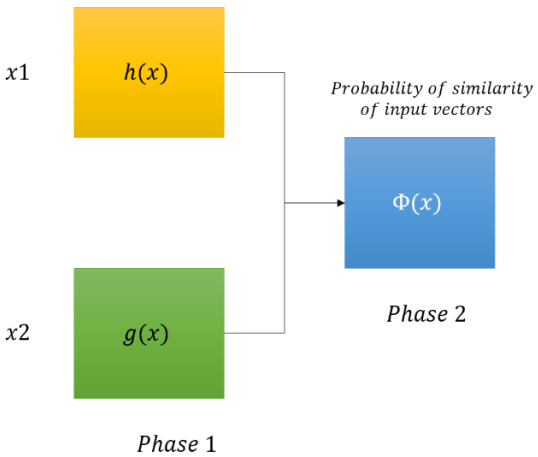


Fig. 2. Mathematical Formulation of the Dual-Phase Framework

Sketch-Based Image Retrieval is a kind of Content-Based Retrieval. The objective is to represent ideas as sketches and identify the most similar images from a collection of images. Because we need to map between drawings and pictures of differing modalities, Sketch-Based Image Retrieval is essentially a Cross-domain challenge [15]. The second stage involves training a model to converge these extracted vectors, which results in the model generating a probability representing the similarity between the sketch and image embeddings.

4.1. Dataset

The study focused on the sketchy database. There are 75,471 sketches of 12,500 objects distributed over 125 categories or classes, and each class has several images. We utilize these sketches and images as a cross-domain correlation since the Sketchy database specifies relationships between pairs of images and sketches. There are also invalid and ambiguous sketches in the database. These have been removed from the database since they are not relevant to our use case. The remaining sketches are into train and test sets with a split of about 10%. As a result, the sketches are divided into four categories: train, test, invalid, and ambiguous. The images that relate to these sketches are likewise organized in the same way. Table 1 shows the distribution of the dataset.

Table 1. Dataset Metrics

Train	Test	Invalid	Ambiguous
58050	7332	885	9214

Train	Test
11209	1291

Despite the fact that there is a significant imbalance in terms of the number of sketches and images, the proposed framework is resilient to this domain imbalance since no pairs are required, and the mapping functions are learned separately.

4.2. Model Architecture

Two VGG16 networks, Convolutional Neural Networks (CNN), are used in the first phase to embed sketches and images as feature vectors. The last layer of the VGG16 network is replaced with a fully connected layer with 125 nodes, one node for each image class. The embedding vectors are extracted from the features vectors of the penultimate layer of these VGG networks, thereby having a dimensionality of 1024 (Fig. 3).

We then pass these features through the phase 2 network for training once they are visualized. The latent/code region is made up of 125 neurons. Each of the other layers (encoder and decoder) has 1024 nodes. The final layers of the Autoencoder Network are compared using the Euclidean distance as a metric.

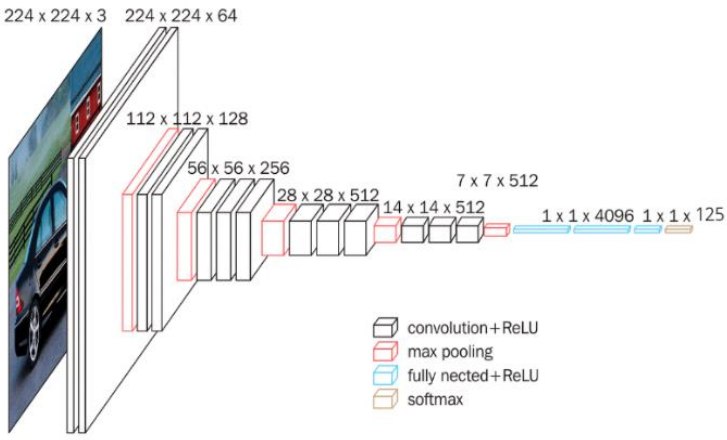


Fig. 3. VGG16 Architecture

4.3. Training

The two VGG16 models are trained on sketches and images independently using a transfer learning-based approach. A pre-trained model was adopted initially. All layers but the last are frozen and trained for 20 epochs. These layers are then unfrozen to enable fine-tuning. The feature vectors of the penultimate layer are extracted. The dimensionality of these vectors is the same as the number of neurons in the penultimate layer - 1024.

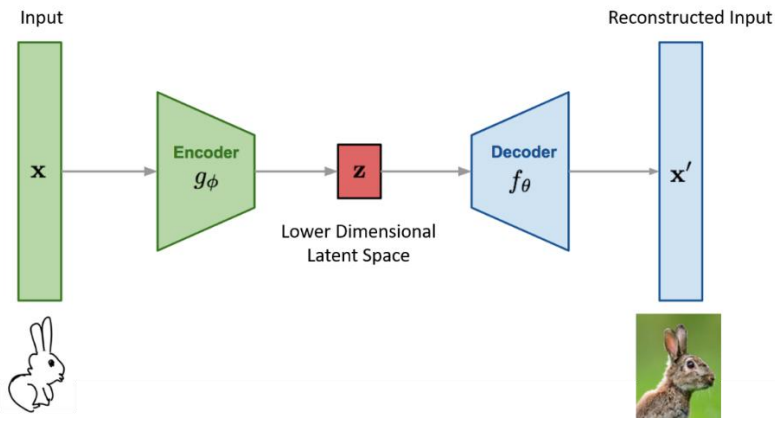


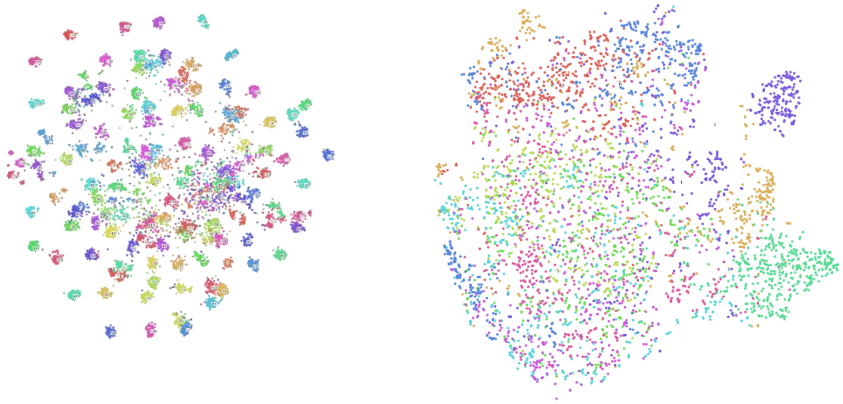
Fig. 4. Autoencoder Architecture

These vector embeddings are analogous to reducing the sketches and images into projections in a 1024-dimensional space. As a result, the VGG16 networks function as dimensionality-reducing processes. We also use a dimensionality reduction approach called t-SNE [17] to visualize them. They use unsupervised learning approaches to model features in higher dimensional space by projecting them onto two-dimensional space, which is graphically represented. In the second phase, the Autoencoder is then trained (Fig.4). Fig. 5 depicts these graphs. The extracted features are paired so that ten pairs of pictures are semantically similar or related for each sketch, i.e. the positive pairings, and 10 pairs of images that are different, i.e., the negative pairs. These pairings are subsequently used to train the Autoencoder, specifically the encoder module, which performs the dimensionality reduction [18, 19].

5.0. Results and Observations

A statistic called Mean Average Precision was utilised to evaluate the retrieval outcomes. Mean average precision is defined as the mean of average precisions returned in the retrieval results for a query. Average precision is determined by plotting a precision-recall curve and then

averaging the value of continuous precision. Mean Average Precision (MAP) [20] quantifies the retrieval results on a sample size of 100.



Training images Training sketches

Fig. 5. TSNE Embeddings

Table 2 shows the training and test accuracy of the VGG16 models used to extract sketch and image features. Table 3 shows the comparison of the performance of the proposed model with baselines. As shown in Table 3, model performs robustly compared with the benchmark scores for MAP@100.

Table 2. Training metrics of Phase 1 models

Model	Training	Validation	Test
Vgg16 Sketch	99.84	71.14	69.17
Vgg16 Image	100	92.37	92.33

Table 3. Result Comparison

Model	MAP@100
Proposed Framework	**0.234**
Siamese Network end to end[1]	0.239
3D Shape[16]	0.192
GF-HOG[10]	0.122

Although the proposed framework doesn't beat the benchmark, it requires lesser computational power and processing time, i.e., the tradeoff between performance and resource/time requirement, makes it a good option for environments with limited computational power.

6.0. Conclusion

Novel Dual-Phase Framework for Cross-domain Content Retrieval in this study. The amount of computational resources required to train a sizeable cross-domain network has significantly reduced. With restricted demand for computing resources, this architecture can help generate powerful results with limited GPU power in a short timescale, producing robust results (although the accuracy may not be the highest obtainable – given the performance and resource requirement tradeoff).

This hypothesis has been verified and tested by training and testing this pipeline on the Sketch-Based Image Retrieval task. The proposed framework outperforms the baselines and has validated the ability of this framework to handle covariate domain shift. Given that this system can generalize effectively to new modalities and domains while offering robust performance, future studies might use it for any cross-domain challenge.

7.0. References

1. Liu, Li, F Shen, Y Shen, X Liu, L Shao, Deep sketch hashing: Fast free-hand sketch-based image retrieval." *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2862-2871, 2017.
2. F Radenovic, G Tolias O Chum, Generic Sketch-Based Retrieval Learned without Drawing a Single Sketch, *ArXiv, abs/1709.03409*, 2007.
3. Lin, Liang, G Wang, W Zuo, X Feng, L Zhang, Cross-domain visual matching via generalized similarity measure and feature learning, *IEEE transactions on pattern analysis and machine intelligence*, 39 (6), 1089-1102, 2016.
4. Shih, J Ling, L H Chen, A new system for trademark segmentation and retrieval, *Image and Vision Computing*, 19 (13), 1011-1018, 2001.
5. Sousa, Pedro, M J Fonseca, Sketch-based retrieval of drawings using spatial proximity, *Journal of Visual Languages & Computing*, 21 (2), 69-80, 2010.
6. C Wang, J Zhang, B Yang, L Zhang, Sketch2Cartoon: composing cartoon images by sketching, *In Proceedings of the 19th ACM international conference on Multimedia*, 789-790, 2011.

7. M J Fonseca, D Gonc alves, Sketch-a-Doc: Using Sketches to Find Documents, *Proceedings of the 16th International Conference on Distributed Multimedia Systems (DMS)*, 327-330, 2010.
8. M J Fonseca, A Ferreira, J A Jorge, Sketch-based retrieval of complex drawings using hierarchical topology and geometry, *Computer-Aided Design*, 41 (12), 1067-1081, 2009.
9. S Liang, Z Sun, B Li, Sketch retrieval based on spatial relations, *International Conference on Computer Graphics, Imaging and Visualization (CGIV'05)*, 24-29, 2005.
10. R Hu, J Collomosse, A performance evaluation of gradient field hog descriptor for sketch based image retrieval, *Computer Vision and Image Understanding*, 117 (7), 790-806, 2013.
11. Y Qi, Y Z Song, H Zhang, J Liu, Sketch-based image retrieval via siamese convolutional neural network, *International Conference on Image Processing (ICIP)*, 2460-2464, 2016.
12. P Sangkloy, N Burnell, C Ham, J Hays, The sketchy database: learning to retrieve badly drawn bunnies, *ACM Transactions on Graphics (TOG)*, 35 (4), 1-12, 2016.
13. Q Meng, D Rueckert, B Kainz, Learning cross-domain generalizable features by representation disentanglement, *arXiv preprint arXiv:2003*, 00321, 2020.
14. H Xia, Z Ding, Structure preserving generative cross-domain learning, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4364-4373 2020.
15. J Wang, Y Song, T Leung, C Rosenberg, J Wang, J Philbin, B Chen, Y Wu, Learning fine-grained image similarity with deep ranking, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1386-1393, 2014.
16. F Wang, L Kang, Y Li, Sketch-based 3d shape retrieval using convolutional neural networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1875-1883, 2015.
17. L V D Maaten, G Hinton, Visualizing data using t-SNE, *Journal of machine learning research*, 9 (11), 2579-2605, 2008.
18. F Radenovic, G Tolias, O Chum, Deep shape matching, *Proceedings of the european conference on computer vision*, 751-767, 2018.
19. H Chen, C Wu, B Du, L Zhang, Deep Siamese Domain Adaptation Convolutional Neural Network for Cross-domain Change Detection in Multispectral Images, *arXiv preprint arXiv:2004*, 05745, 2020.
20. K Oksuz, B C Cam, E Akbas, S Kalkan, Localization recall precision (LRP): A new performance metric for object detection, *Proceedings of the European Conference on Computer Vision*, 504-519, 2018.