Augmented TIRG for CBIR Using Combined Text and Image Features

Mohamed Aboali
Computer and Systems Engineering
Ain Shams University
Cairo, Egypt

Islam Elmaddah
Computer and Systems Engineering
Ain Shams University
Cairo, Egypt

Hossam El-Din Hassan
Computer and Systems Engineering
Ain Shams University
Cairo, Egypt

Abstract*— in this paper we propose a methodology for Content Based Image Retrieval, CBIR, using query inputs in the form of a source image and text modifiers. The proposed methodology augments the methodology proposed in [21], TIRG, with a trained module. The trained module aims at enhancing the relationship between a) the composed image-text features and b) the target image features (e.g. input an image of blue dress along with a textual description and ask for the same dress but in red). Our study used two trained modules (Linear Regression, LR, and Non-Linear Multilayered Perceptron, NMLP). The proposed models were tested using the well-known fashion 200K dataset. The LR model reduced the Mean Squared Error, MSE, significantly. A joint LR model outperformed the TIRG on the testing Dataset. Two NMLP trained models were used: MSE-optimized and Cosine similarity optimized. The performance of the two models was very similar. The NMLP models, in general, outperformed TIRG over the Training dataset. The study also indicates that combining image-text features should be kept to later stages to obtain their recall intersections. Moreover, the study showed that text-features generation based on words assigned numbers irrelated to their semantics requires semantic hub to bridge to a semantic-numbers.

I. INTRODUCTION

Computer vision is one of the leading research fields nowadays due to the wide range of its possible applications as a result of the recent advances in storage, processing, and data transfer technologies. Computer vision includes methods for acquiring, enhancing, restoring, analyzing, recalling, encoding, and understanding of digital images. The process mainly aims at the extraction of high-level information from real-world images to produce numerical or symbolic information, that will likely be in the forms of decisions. Understanding vision in this context means the transformation of visual images (as the input of the human retina) into descriptors that make sense to a computer system and can elicit appropriate action. One of scientific bases of Computer Vision is building an intelligent machine-based storing and recalling of images in unconventional ways.

The usage of images, specifically digital ones, in our daily life increases exponentially by time. The handy mobile digital cameras and other image acquisition cheap devices made it easy to capture high-resolution images. The cheap high-capacity memory made it possible to store large number of images on handy devices. Images carry out more information as factually supported by the well-known quote "image is better than one thousand word". The Internet capabilities for transmitting and sharing images added a new dimension to the use of images. Cloud storage and computing capabilities made it possible to process and store massive

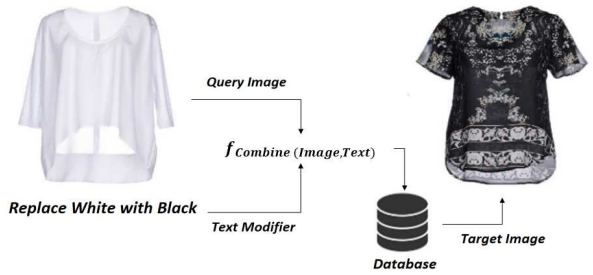

Fig. 1. CBIR example with the inputs are image and text

amounts of high resolution images. Image acquisition, processing, and storing are not only getting cheaper but also better with time. This technological trend seems to continue for decades to come. The main problem, now, is not in storage or acquisition, the problem lies in recalling from a large dataset of images. Without an effective recall, searching in images will be a painful process or perhaps even useless waste of storage.

Content Based Image Retrieval, CBIR, is a specialized retrieval technique for images. Images are two-, or three-dimensional data that represents a set of objects in a context or in an interaction. The realization of computer vision mandates the ability to recall images based on similarity, and/or descriptive attributes. As an example: a doctor has a chest X-Ray of a COVID-19 patient (an image) and would like to see similar successful treatment tracks based on that image and possibly some patient descriptive data. Here comes the role of CBIR. Fig. 2 depicts an overview of CBIR architecture.

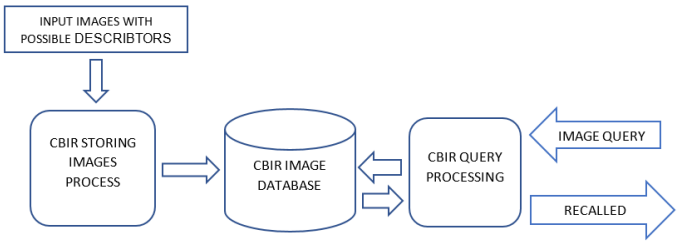

Fig. 2. Shows CBIR process, Query, Storing images in CBIR Database

An image comparative is neither a picture element nor pixels colors comparison process as these change drastically by scale, rotation, translation, illumination, pose, orientation, displacements and many other factors [2]. The former mentioned changes have nothing to do with images objects, objects interactions or information learned from the seen. Therefore, Images similarities metrics are not applied on raw images data but rather on images features extracted through image processing. Features are high-level abstraction of image contents. Also, image query formation is not as easy as data

978-1-6654-4231-2/21/$31.00 ©2021 IEEE

predicates as it requires image as one of the potential inputs with some applied modifiers. The recalled images, in general, have high level of uncertainty. Therefore, possible multiple images recall, or more than one recall cycle is the default pattern.

Image retrieval based on content has numerous application domains including, but not limited to, fashion, graphic design, games, simulations, publishing, advertising, historical surveys, architectural engineering, crime prevention, medical diagnosis, geographical information, and remote sensing systems [3]. A typical image retrieval application example is in the clinical decision-making process; it is critical and important to find other images of the same modality and the same anatomic region of the same disease. Clinical decision support techniques such as case-based reasoning or evidence-based medicine can even produce a stronger need to retrieve images that can be valuable for supporting certain diagnoses.

In this paper, a CBIR approach is proposed to enable the recall of images using a textual description as a modifier for image descriptors. The approach uses neural networks as feature extraction vehicle from both images and text descriptors. The networks used are deep convolutional network for images and recurrent neural networks for text. The selected networks are ResNet and LSTM for the former tasks respectively [5][6]. The selected networks proved to outperform many others [7][8].

The rest of the paper is organized as follows. Section 2 is the related work, Section 3 an overview of content-based image retrieval. Section 4 convolution neural networks, recurrent neural networks, ResNet and LSTM overview. Section 6 presents the proposed model. Section 7 tests and results. And finally, Section 8 study conclusion and future work.

II. Related work

Our work is heavily influenced by two basic principles in CBIR, the principle of composition and the use of neural networks in feature extraction. The composition principle, in broad view, has its historical roots in philosophy, neuroscience, mathematics, and many of computer science domains such as natural language understanding, sensors fusion, visual recognition, etc. In visual recognition, which is the focus of this research, the principle objective is to find out new concept from primitive elements. The principal base is, in general, a statistical learning from samples. This opens the door for models training using samples [21][32] to produce the new, composed, concepts. The compositionality for visual recognition researches includes Biederman's Recognition-By-Component's theory [33] and Hoffman's part theory [34].

The use of convolutional Neural Networks , CNN, as a mean of feature representations of multiple semantic is used in [32]. Objects composition for recognition systems uses Deformable Part Models in [35], grammars in [36], and AND-OR graphs in [37]. Composition as pivotal element is used for visual question answering in [38], handwriting recognition in [39], and zero-shot detection in [40]. A lot of research has been done to improve features composition retrieval performance by user's feedback on relevance [21]. In [28], the text feature is incorporated by mapping into parameters of a fully connected layer within the image CNN. Another domain that incorporates text images composition is visual question answering [42]. The residual connection used to enforce composition in [43]. In [44] recurrent model to colorize images given text descriptions. In [21] composition classifier was trained to combine an object classifier and attribute classifier. The attribute embedding operator was used in [26].

III. Content Based Image Retrieval

The term "Content-Based" refers to analyzing and attempting to comprehend the contents rather than the use of metadata such as keywords, and descriptions associated with the object/concept. "Content" of images implicitly refers to colors, shapes, textures, or even objects interactions inferred from the image. Field researchers refers to information extracted to represent these contents by "features". Query formation, features extractions, combination of features and distance metrics are the main research topics of the CBIR [9][10] and are discussed further in the following subsections.

A. Query Formation

In general, query formation is driven by the application needs as well as the form of data or information available for the query. The specification of such queries, in case of CBIR, is significantly difficult as it relates to abstract hypothetical images or thoughts formed within users' mind. The expression of such query requires widening the means. Generally, for CBIR, there are several types of query formation used by researchers, such as: keywords, images, sketch maps, color maps, context maps, canvases, and spatial icons [11]. Query by keywords requires text annotations for images collected in a database and is to date used in Internet search engines. Some treatment is needed for ambiguity resolution as well as minimizing the gap between the image labeling and the image itself [12]. The annotation process for large database is painful and likely insufficient and lakes the precision. Query by example (QBE): in which a set of attributes or features hopefully describe the contents of the user's desired image [13][14] which the user needs to find. Query by canvas enables users to use composition of geometrical shapes, colors, and textures. The canvas basically express the needs using primitive features [13][14][15]. Query by spatial icons using a higher-level visual semantics represents the spatial arrangement of image segments. Combinations of the former mentioned formations are also used in [12] [16][17]. Text as well as example image are also used in [18]. In this research, the text features extracted from the text, rather than keywords, associated with images combined with extracted image features to form combined images features to use in learning recall images from databases.

B. Feature extraction

Is a focal point in the success of any computer vision systems. Feature extraction could be engineered, or machine learned. In case of engineered features, the designer studies the features available and likely an experimentation study is made over sample set to select a set of features that is the features selection. Then, an algorithm is designed for the extraction process of the selected features. This engineered approach assures that the selected features carry the desired properties which ensures the detection of insignificant distances for similar images and vice versa. Moreover, small variance for scale, rotation, noise, pose, translation adds a very low computational cost [19]. The extraction process, likely, include pre-processing as well as the extraction algorithm.

In machine learning feature extraction, the machine vehicle is trained to extract the features. The main vehicle used for that is artificial neural networks [20] where the network is engineered to suit the task and a training dataset is also

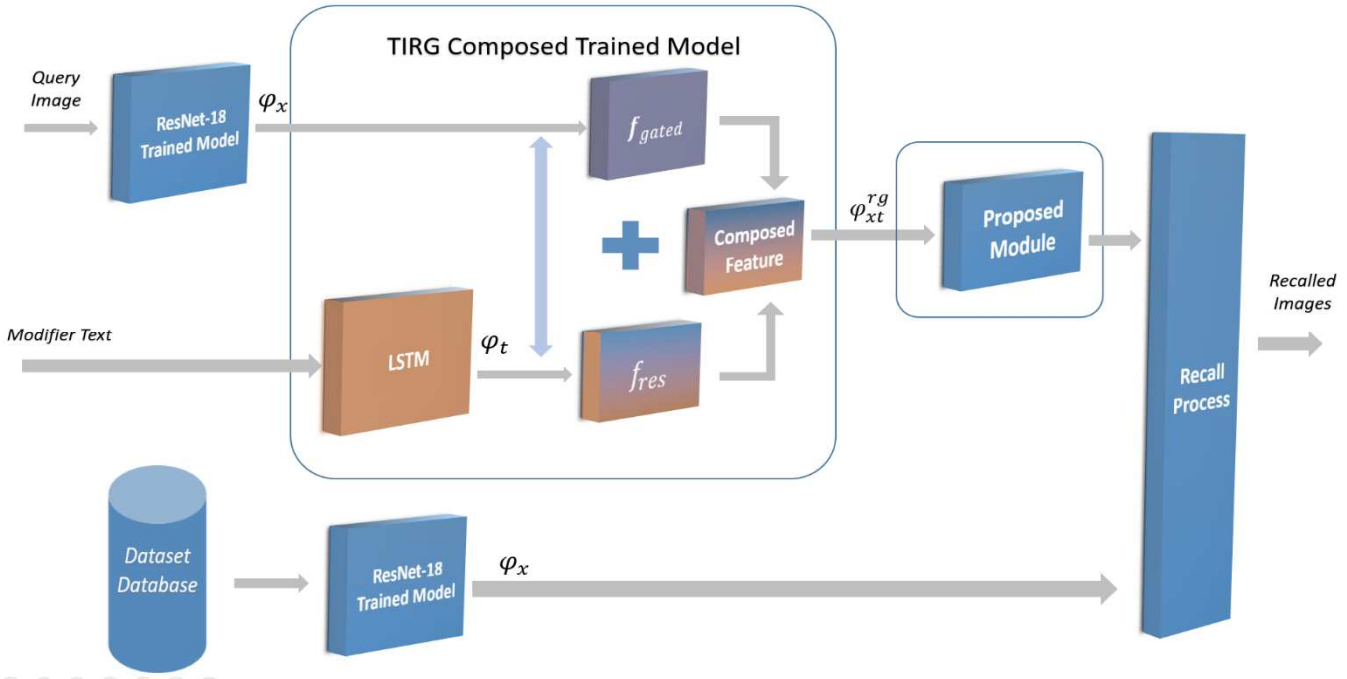

Fig. 3. The CBIR using combined query Image and modifier text uing TIRG and our proposed module.

engineered. Then, network is trained and validated using the prepared training dataset. The process seems easier than the engineered feature extraction approach, however, there are many questions the designers must answer such as: which network architecture to use? How many layers? How many neurons in each layer? What is the proper training and validation dataset?

Image features bases include color, texture, and shape. Color features include: histogram, Correlogram, Auto-Correlogram, Coherence vector, and the dominant Color descriptors [19]. Texture features include co-occurrence matrix, Gabor Filter, Wavelet transform, and Tamura. The shape features include moments, Fourier descriptors, discrete cosine descriptors, principal components analysis (PCA), Multidimensional Scaling (MDS), Canny algorithm, and SIFT descriptors [21].

Feature learning or automatic representation is embedding machines with basic set of techniques which allows systems to automatically discover features from raw data. That is feature learning is engineered rather than extracted. Feature learning can be either supervised or unsupervised. Supervised learning requires training and validation data to be labeled. also requires the supervised networks perceptron, radial bases, and Convolution neural networks. In unsupervised learning, features are learned without the need for labels. But from the unsupervised architectures networks such as Kohonen network, self-organizing map, and Hopfield network.

Feature vectors contains measurements of different attributes of images or image-objects. Therefore, Feature vectors, normally, requires preprocessing before using due to the heterogonous nature of its elements. Standardization and normalization could be part of the extraction process for both engineered or learned cases. Features could be extracted either in a local or global fashion. In global features extraction the feature operator applies to the whole image. Local features extraction requires search regions of interest to apply the local operator[19].

C. *Features vectors combination*

Is needed when different sources of features exist such as case of text and image. This problem was a topic of research, too. In [21] a Text Image Residual Gating, TIRG, function is used combine image and text features. The TIRG function is weighted sum of Residual function and gating function. The weights supposed to be adapted by neural like learning process. The core of both residual and gating functions is a sequence of convolution, Relu, and convolution on text features broadcasted on image features. The gating function has two more steps sigmoidal application followed by dot product by image features. The author's intent to "modify" the query image feature instead "feature fusion" to create a new feature from existing ones. Multilayer perceptron is used to concatenate text and image features in [22][23][24]. LSTM, as recurrent model, fed by image features followed by text words used in [25]. Text is used to form transfer matrices for image features in [26]. Visual question answering methodologies, which focus on finding answer to natural language question on a given image [27], also were used in [28-30].

D. *Similarity Metrics*

Content-based image retrieval (CBIR) not only needs efficient extraction of features but also an effective similarity metric to measure the distance between features vectors. Various distance metrics are used to measure distance between vectors which include Euclidean distance, city block distance, Canberra distance, maximum value metric, Minkowski distance, Mahalanobis Distance, Histogram Intersection Distance, and Quadratic Form Distance [31]. Distance metrics are characterized by their accuracy and the time complexity.

IV. Convolution and Recurrent Networks

A. Convolution Neural Networks

Are deep feed forward layered networks. Input images with possible rescaling and normalization applications constitute the input neurons. Convolution kernel filters used

to highlight features throughout images. Using different kernel means looking for more features. Stride option provides allowance to convolution filters to skip pixels as features detected likely be redundant and at same time reduce computational complexity. Padding facility controls the size reduction of images as the process goes on due to boundary limits on images. Pooling functions remove duplicate or insignificant features as well as allowance of features translation. The convolution and pooling layers perform feature extraction through making features map. Convolution layers are followed by traditional feed forward neural network layers for classification, recognition, or any other function based on extracted feature map. Convolution architecture represents the base for variety of networks. In recent years, Large Scale Visual Recognition Challenge (ILSVRC) was a reference to measure the progress of computer vision for large-scale image indexing for retrieval and annotation as well as object detection and image classification at large scale. The winner of the contest in 2011 was SIFT little later, starting from 2013, were convolution-based architectures specifically AlexNet, ZFNet, SPPnet, PNASNet, and ResNet several architectures.

ResNet architecture is a convolution network characterized with skip or shortcut connection [5]. This feature avoids the loss of information at later layers of the network because of the operators of the earlier ones. ResNet most famous architectures are RestNet18, ResNet34, ResNet50, ResNet101, and ResNet152. The architecture could be a blocks-of-layers based on a basic building block.

B. Recurrent Neural Networks and LSTM

Judging an event highly depends on its context. Understanding word in the middle of a sentence requires comprehending the previous words and, in some cases, requires memories from even previous sentences. Non-recurrent neural networks, traditional neural networks, cannot do that. The recurrent architecture as it goes through states according to the current state and the current input, The recurrent neural networks applications include Machine Translation, Robot Control, Time Series Prediction, Speech recognition, Speech Synthesis, Optimization, and Text Embedding. The architectures of recurrent neural networks include Hopfield, GRU, and LSTM. The Long-Short Term Memory, LSTM, network, as its structure, can carry concepts from long back in iterations as well as significant learning parameters.

V. Proposed Methods

The proposed model, Fig. 3, is based on the model introduced in [21]. The base model uses ResNet18 and LSTM networks for image and text features extraction followed by the Text Image Residual Gating function, TIRG[21]. The ResNet18 architecture used is a pre-trained 'torch library' model, without the fully connected layer, to extract image features, φ_x, The LSTM network is used to extract text features, φ_t, The TIRG function is used for text and image features composition. The main equation in [21]:

$$\varphi_{xt}^{rg} = \omega_g \, f_{gate}(\varphi_x, \varphi_t) + \omega_r \, f_{res}(\varphi_x, \varphi_t) \quad (1)$$

Where φ_{xt}^{rg} represent TIRG, ω_g , ω_r are learning weights, φ_x represent the last layer of ResNet 18 (ResNet 17) , Last convolution layer, (H*W*C), W is the width, H is the height, and C = 512 is the number of feature channels of image, φ_t Represent the Last layer of LSTM of text (C= 512), the following $f_{gate}(\varphi_x, \varphi_t)$, $f_{res}(\varphi_x, \varphi_t)$ are two linear mapping convolution functions called gating and residual functions.

$$f_{gate}(\varphi_x,\varphi_t) = \sigma(W_{g2} * \text{RELU}\,(W_{g1} * [\varphi_x,\varphi_t])) \odot \varphi_x \quad (2)$$

Where σ is a sigmoid function, W_{g2} W_{g1} 3*3 are convolutional filters, $[\varphi,\varphi_t]$ are broadcast φ_t over φ_x , $\odot$ is an element product

$$f_{res}(\varphi_x,\varphi_t) = W_{r2} * \text{RELU}\,(W_{r1} * ([\varphi_x,\varphi_t])) \quad (3)$$

The TIRG function is a typical layer of deep network followed by a fully connected layer. Therefore, the learning TIRG architecture is trainable to adapt to the training set. The proposed model Fig. 3 is based on considering TIRG a text-image composition methodology. The trained module will add more optimization to relation between φ_{xt}^{rg} of the query composed image and text and φ_x of the target image. The proposed added modules are a) linear regression and b) non-linear perceptron.

A. Linear Regression

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable. In essence, multiple regression is the extension of ordinary least squares (OLS) regression because it involves more than one explanatory variable. In the multiple regression setting, because of the potentially large number of predictors, it is more efficient to use matrices to define the regression model and the subsequent analyses; in this case, we will add a matrix formulation of the multiple regression model [4] after we extract TIRG features. The model is in the form:

$$\begin{bmatrix} y_1 \\ y_1 \\ . \\ . \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ & . \\ & . \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_1 \\ . \\ . \\ \varepsilon_n \end{bmatrix}$$

$$Y = X\beta + \varepsilon$$

Where Y is the target feature extracted from the last layer of ResNet18 (ResNet17) of size 512, X is TIRG feature of size 512, β represent the Slope of coefficient, ε the model error term (known as residuals), β Will be generated in the form:

$$\beta = ((X' * X_{transpose})^{-1} * X_{transpose}) * Y$$

Where X' represent TIRG Feature with 1 added for basis in the First column of the Matrix, β will be a Matrix of 513 * 512, our equation will be $Y_{512} = X'_{513} * \beta_{513*512}. + \varepsilon$ Our two regression models are generated from training dataset and the joint training and testing datasets.

B. Non-Linear Multi-layered Perceptron

The architecture used in NMLP network includes two hidden layers. The model built to expand and reconstruct

features through the hidden layers as input and output layers' size is the same, 512 neurons. The architecture uses input first hidden linear (512, 1050), followed by sigmoid function and features expansion function. The first to second hidden layer is contract linear layer (1050, 950). Finally, another contraction layer from the second hidden to the output linear layer (950, 512). Optimizing the network requires defining loss function, learning rate and batch size. In the optimization, two different loss functions were used: MSE and Cosine Similarity Loss function: $loss$ =abs (1-COS (Target Vector, Output Vector)). The result of two models were similar the COS similarity was slightly better as the final recall metric is Cosine similarity. Learning rate used was 0.0015. For sake of space, we will only present the optimized performance of the Cosine similarity.

The NMLP performs an additional mapping in vectors domain without change in the vector dimensionality. Perceptron Network mapping, $\|\varphi_{TRG}\| \mapsto \varphi_p$ where $\|\varphi_{TRG}\|$ is the unit vector transformation and φ_p is the features vector produced by the perceptron networks. The perceptron network learns the transformation during the training phases using the training data set. After training, the network will be ready for the operational mode.

VI. Experimental Results

The proposed models were tested on the well-known Fashion200k. The dataset contains 200K images from different classes of fashions items. Each image of the dataset comes with brief text about the item, a text modifier and a target image id. Our quarries created in the same way as [21], The modification text is considered our query. For the sake of comparison, training and testing datasets are formed similarly. The Training set formed from 172K pairs and testing was conducted using 33K pairs.

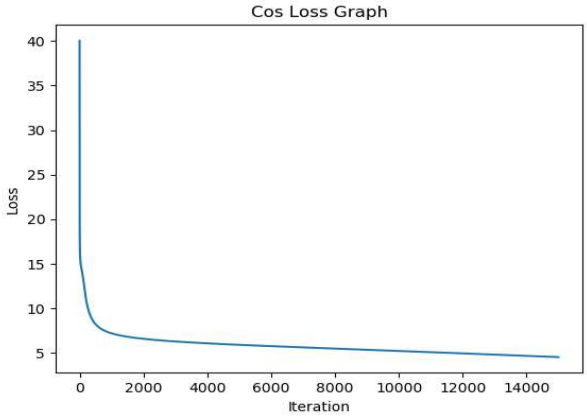

Fig. 4. learning curve for COS optimized NMLP network

TABLE 1. TIRG RESULTS IN COMPARISON WITH OTHERS.

Method	R@1	R@10	R@50
Han et al	6.3	19.9	38.3
Image only	3.5	22.7	43.7
Text only	1.0	12.3	21.8
Concatenation	11.9±1.0	39.7±1.0	62.6±0.7
Show and Tell	12.3±1.1	40.2±1.7	61.8±0.9
Relationship	13.0±0.6	40.5±0.7	62.4±0.6
MRN	13.4±0.4	40.0±0.8	61.9±0.6
FiLM	12.9±0.7	39.5±2.1	61.9±1.9
TIRG	14.1±0.6	42.5±0.7	63.8±0.8

The trained models, based on [21], for TIRG network to compare the performance and build the model. The model best performance reported in [21] is summarized in Table 1. The LR model using the training dataset performance on the testing dataset is outlined in Table 2. The joint LR training and testing data set performance on testing dataset is listed in Table 2.

TABLE 2. LR PERFORMANCE

Model	R@1	R@10	R@50
TIRG	14.1±0.6	42.5±0.7	63.8±0.8
LR-training	12.2±0.2	38.6±0.31	60.8±0.5
LR- joint	22.7±0.9	57.6±0.4	76.5±0.6

MSE of the training dataset before LA is 121.27 and after 4.80, for testing dataset MSE before LA is 119.7 but after 5.45. However, from the table above we can easily see that there is a significant reduction in MSE on the training dataset, the performance on the testing dataset is below TIRG's.

The NMLP perceptron network trained on the training data set for 15K iterations model snapped every 1K iteration. Fig. 4 show the learning curve for Cosine similarity optimized network. The learning curve shows a significant reduction of the total loss from over 40 to less than 4. The recall performance of the network coincides with this enhancement in training loss. The performance of the network as training goes is enhanced greatly on the training and validation datasets. Table 3. summaries the performance as the training goes based on the snapped models during the training.

TABLE 3. NMLP PERFORMANCE AS THE TRAINING GOES ON

Model	R@1	R@10	R@50
TIRG -train	33.2±0.3	73.1±0.7	93.1±0.4
TIRG -test	14.1±0.6	42.5±0.7	63.8±0.8
NMLP-1K-train	32.5±0.5	67.2±0.3	88.1±0.1
NMLP-1K-test	11.4±0.3	37.7±0.2	59.1±0.7
NMLP-4K-train	42.4±0.4	75.1±0.4	92.1±0.7
NMLP-4K-test	10.5±0.63	37.5±0.3	59.1±0.6
NMLP-8K-train	49.1±0.2	79.9±0.4	94.08±0.3
NMLP-8K-test	10.3±0.3	36.2±0.6	59.9±0.5
NMLP-15K-train	54.6±0.4	83.6±0.4	95.5±0.7
NMLP-15K-test	10.2±0.63	35.6±0.3	59.1±0.6

From the table we can conclude that as training goes the performance is significantly enhanced on the training dataset over TIRG's. However, on the testing dataset performance remains below and even get little worse as training goes on.

VII. Conclusion

In this paper, a proposed text and image composition model for CBIR was introduced. The model based on the TIRG function proposed in [21]. We used a pre-trained updated version of the TIRG model augmented with trainable module. Two trained modules were used in this study, LR and NMLP. The fashion 200k dataset was used in testing the proposed models. LR regression model using training dataset enhances the MSE but the performance on testing dataset was below the TIRG. LR using the joint training and testing

datasets outperforms the TIRG on testing dataset. The NMLP model greatly outperforms the TIRG on the training dataset, but that enhancement was not reflected on the testing dataset performance. The study points to the fact that early composition or compression of image and text features leads to early-undesired losses. Therefor both text and image features should be kept later stages to even, likely, getting the intersections of the two recalls from the image database. Moreover, text features based on numbers allocated to words where these numbers are loosely coupled to the associated semantic leads to, away from the training set, performance degradations. Therefore, a kind of ***semantic hub*** is needed to bridge these numbers to possible semantic-features or semantic-numbers.

REFERENCES

[1] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: A tutorial and survey," IEEE Journal of Solid-State Circuits, vol. 41, no. 3, pp. 712–727, Mar 2006.

[2] Kavita. R. Singh , Mukesh. A. Zaveri, Mukesh. M. Raghuwanshi ," Illumination and Pose Invariant Face Recognition: A Technical Review", International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM), ISSN: 2150-7988 Vol.2 (2010), pp.029-038.

[3] Jing Li, "The application of CBIR-based system for the product in electronic retailing," 2010 IEEE 11th International Conference on Computer-Aided Industrial Design & Conceptual Design 1, Yiwu, China, 2010, pp. 1327-1330.

[4] Scott H. Brown, Multiple Linear Regression Analysis: A Matrix Approach with MATLAB

[5] Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun Deep Residual Learning for Image Recognition

[6] a tutorial into Long Short-Term Memory Recurrent Neural Networks Ralf C. Staudemeyer, Schmalkalden,2019

[7] Evaluating the Performance of ResNet Model Based on Image Recognition, Riaz KhanXiaosong Zhang,2018

[8] A review on the long short-term memory model Greg Van Houdt1 · Carlos Mosquera2 · Gonzalo Nápoles1, 2020

[9] A Decade Survey of Content Based Image Retrieval using Deep Learning, Shiv Ram Dubey, Member, IEEER

[10] Analysis of distance metrics in content-based image retrieval using statistical quantized histogram texture features in the DCT domain Author links open overlay panel,Fazal Malik, Baharum Baharudin

[11] Tatiana Jaworska, Query techniques for CBIR, Systems Research Institute, Polish Academy of Sciences, 6 Newelska Street, Warsaw, Poland, 2016.

[12] Wang, X., Liu, K. & Tang, X., 2011. Query-Specific Visual Semantic Spaces for Web Image Re-ranking.. s.l., s.n., pp. 1-8.

[13] Niblack, W. et al., 1993. The QBIC Project: Querying Images by Content Using Colour, Texture and Shape. SPIE, Volume 1908, pp. 173-187.

[14] Rubner, Y., Tomasi, C. & Guibas , L. J., 2000. The Earth Mover's Distance as a Metric for Image Retrieval. International Journal of Computer Vision, 40(2), pp. 99- 121.

[15] Ogle, V. E. & Stonebraker, M., 1995. CHABOT: Retrieval from a Relational Database of Images. IEEE Computer, September, 28(9), pp. 40-48.

[16] Lim, J.-H. & Jin, J. S., 2005. A structured learning framework for content-based image indexing and visual query. Multimedia Systems, Volume 10, p. 317–331.

[17] Rasiwasia, N., Moreno, P. J. & Vasconcelos, N., 2007. Bridging the Gap: Query by Semantic Example. IEEE TRANSACTIONS ON MULTIMEDIA, Aug, 9(5), pp. 923- 938.

[18] Nam Vo, et. al., "composing text and image for image retrieval – an empirical odyssey", IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[19] A review on image feature extraction and representation techniques, TianDongping Tian,2013

[20] Research on image classification model based on deep convolution neural network Mingyuan Xin and Yong Wang,2019

[21] Nam Vo, et. al., "composing text and image for image retrieval – an empirical odyssey", IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[22] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. VQA: Visual Question Answering. In ICCV, 2015.

[23] X. Guo, H. Wu, Y. Cheng, S. Rennie, and R. S. Feris. Dialog-based interactive image retrieval. arXiv preprint arXiv:1805.00145, 2018

[24] H. Nam, J.-W. Ha, and J. Kim. Dual attention networks for multimodal reasoning and matching. In CVPR, 2017.

[25] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In CVPR, 2015

[26] T. Nagarajan and K. Grauman. Attributes as operators. 2018.

[27] Qi Wu, et. al. " Visual question answering: A survey of methods and datasets", Computer Vision and Image Understanding journal, Volume 163, October 2017, Pages 21-40

[28] H. Noh, P. Hongsuck Seo, and B. Han. Image question answering using convolutional neural network with dynamic parameter prediction. In CVPR, 2016.

[29] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In NIPS, 2017.

[30] J.-H. Kim, S.-W. Lee, D. Kwak, M.-O. Heo, J. Kim, J.-W. Ha, and B.-T. Zhang. Multimodal residual learning for visual qa. In NIPS, 2016.

[31] Amit Singh, et. al. "A Hybrid Approach for CBIR using SVM Classifier, Partical Swarm Optimizer with Mahalanobis Formula", International Journal of Computer Applications (0975 – 8887) Volume 111 – No 12, February 2015

[32] Ishan Misra, Abhinav Gupta, and Martial Hebert," From Red Wine to Red Tomato: Composition with Context ", IEEE Conference on Computer vision and pattern recognition (CVPR), Honolulu, Hi, 2017, pp. 116—1169.

[33] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Neural module networks. In CVPR, pages 39–48, 2016.

[34] D. D. Hoffman and W. A. Richards. Parts of recognition. Cognition, 18(1):65–96, 1984.

[35] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8. IEEE, 2008.

[36] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image parsing: Unifying segmentation, detection, and recognition. International Journal of computer vision, 63(2):113–140, 2005.

[37] Z. Si and S.-C. Zhu. Learning and-or templates for object recognition and detection. IEEE transactions on pattern analysis and machine intelligence, 35(9):2189–2205, 2013.

[38] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Learning to compose neural networks for question answering. In NAACL, 2016.

[39] B. M. Lake, R. Salakhutdinov, J. Gross, and J. B. Tenenbaum. One shot learning of simple visual concepts. In Proceedings of the 33rd Annual Conference of the Cognitive Science Society, volume 172, page 2, 2011.

[40] V. Krishnan and D. Ramanan. Tinkering under the hood: Interactive zero-shot learning with net surgery. arXiv preprint arXiv:1612.04901, 2016.

[41] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. arXiv preprint arXiv:1606.01847, 2016.

[42] D. Teney, P. Anderson, X. He, and A. van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In CVPR, 2018.

[43] A. Miech, I. Laptev, and J. Sivic. Learnable pooling with context gating for video classification. arXiv preprint arXiv:1706.06905, 2017.

[44] J. Chen, Y. Shen, J. Gao, J. Liu, and X. Liu. Language-based image editing with recurrent attentive models. In CVPR, 2018.