Real Data Augmentation for Medical Image Classification

Chuanhai Zhang[1], Wallapak Tavanapong[1], Johnny Wong[1], Piet C. de Groen[2], JungHwan Oh[3]

[1]Department of Computer Science, Iowa State University, Ames, IA, USA

[2]Department of Medicine, University of Minnesota, MN, USA

[3]Department of Computer Science and Engineering, University of North Texas, TX, USA

Email: {czhang89, tavanapo, wong}@iastate.edu, degroen@umn.edu, Junghwan.Oh@unt.edu

Abstract. Many medical image classification tasks share a common unbalanced data problem. That is images of the target classes, e.g., certain types of diseases, only appear in a very small portion of the entire dataset. Nowadays, large collections of medical images are readily available. However, it is costly and may not even be feasible for medical experts to manually comb through a huge unlabeled dataset to obtain enough representative examples of the rare classes. In this paper, we propose a new method called Unified LF&SM to recommend most similar images for each class from a large unlabeled dataset for verification by medical experts and inclusion in the seed labeled dataset. Our real data augmentation significantly reduces expensive manual labeling time. In our experiments, Unified LF&SM performed best, selecting a high percentage of relevant images in its recommendation and achieving the best classification accuracy. It is easily extendable to other medical image classification problems.

Keywords: Real data augmentation, unbalanced data, image classification.

1 Introduction

To use supervised machine learning in the medical domain, highly skilled expertise is required to create a training dataset with sufficient representative images for all the classes. Data imbalance is prevalent due to two major factors. For a given disease of interest, there are more healthy patients than unhealthy ones. For a given patient, typically there are more normal images than the abnormal ones. For instance, in a colonoscopic procedure, most frames showing normal colon mucosa compared to no frames or a few minutes of frames showing a polyp and a snare for polypectomy.

Traditional data augmentation is commonly used to address the data imbalance problem [1, 2]. This approach applies image processing operators such as translation, cropping, and rotation on images in the training dataset to create more images for the classes with fewer labeled samples. However, the limitation is that, depending on the parameters and image operators used, the generated samples may not represent image appearances in real data or the generated samples may be very similar to the existing

images in the training dataset. Random data dropout addresses the data imbalance problem by randomly dropping out data of the class with many more examples (e.g., the normal class) [3]. However, this method does not increase the learning capability for the rare classes.

We investigate a different paradigm that selects images from a large unlabeled dataset and recommends them to the medical expert. We call this paradigm "real data augmentation" since the recommended images are from a real dataset. One naive real data augmentation method is to select images from the unlabeled dataset randomly without replacement and ask the medical expert to assign them class labels. This approach is time-consuming and costly to obtain enough representative examples of the rare target classes. On the other hand, a self-training method [4] applies a probabilistic classifier trained on the seed labeled dataset to predict the class of each unlabeled image and recommend for each class the images with the highest probabilities of belonging to that class. However, the low classification accuracy caused by the small training dataset likely results in incorrect recommendations. Some real data augmentation methods were introduced for text classification [5] and object recognition [6]. These methods use two steps. First, the feature representation is learned. Then, a fixed distance function, (e.g., the L2 distance, the cosine similarity), is used to retrieve relevant samples.

Our contribution in this paper is as follows. (1) We propose a new real data augmentation method called **U**nified **L**earning of **F**eature Representation and **S**imilarity **M**atrix (Unified LF&SM) using a single deep Convolution Neural Network (CNN) trained on the seed labeled dataset. The method recommends top k similar images to the training images for each class to augment the seed dataset for that class. (2) We explore two more real data augmentation methods, the two-step method that learns feature representation first then learns the similarity matrix later and the method that learns only feature representation using a fixed similarity function. (3) We evaluated the effectiveness of the three methods and the self-training method. The effectiveness is in terms of the number of relevant images in the top k recommended images and the classification accuracy for the problem of 6-class classification of colonoscopy and upper endoscopy images. We found Unified LF&SM most effective among the four methods in our experiments.

2 Methods

We describe four methods for real data augmentation in this section. They differ in how feature representations are obtained and the recommendation algorithm to select unlabeled images. Let T be a labeled training image dataset, N_C be the number of classes desired for the classification problem, and N_j be the number of images in T belonging to a class j. Let $\mathcal{U}$ be an unlabeled dataset with a cardinality of N_s. Our goal is to recommend the set (R_j) of k most relevant images from $\mathcal{U}$ for each class j. We use CNN as our supervised deep learning classification algorithm. In this paper, we investigate the simplest recommendation algorithm, which recommends the top k most similar images for each class to improve the robustness of CNN. The higher the value

of k is, the larger the variation in the recommended examples is. Note that even the most similar image is recommended, it is still useful since the image is from a different video never seen in the training set.

2.1 Data augmentation based on probabilities (CNN + Probability)

After training a CNN classifier on T, we apply the classifier to each image I_i in $\mathcal{U}$ and obtain the corresponding value $p_{(i,j)}$ indicating the probability of the image I_i belonging to a class j using the soft-max function at the last layer of the CNN. Fig. 1 shows the recommendation algorithm. The structure of the CNN we used is described in Section 3.1.

```
for j = 1, 2, ..., N_C
  for i = 1,2, ..., N_S
    Compute p_(i,j) using the CNN classifier
  end for
  Sort p_(1:N_s, j) in descending order /* sort images based on probability */
  Assign top k images to the set R_j for the class j
end for
Return R_j for each class j
```

Fig. 1. Recommendation algorithm---"CNN + Probability"

2.2 Data augmentation based on distance function learning (CNN + Bilinear)

We train a CNN classifier on the training dataset T. Then we extract the feature representation v_i for the image I_i using the trained CNN. Next, we apply OASIS [7] to learn a bilinear similarity function $S_W(v_i, v_j)$ in Equation 1 that assigns higher similarity scores to images in the same class. Fig. 2 shows our method based on the bilinear similarity function to find similar images.

$$S_W(v_i, v_j) = v_i^T \, W \, v_j \tag{1}$$

```
for j = 1, 2, ..., N_C
  Compute the center of the feature vectors of all images of
  the class j in T: v̄ = Σ_{i=1}^{N_j} v_i / N_j
  for i = 1, 2, ..., N_S
    Compute v_i^T W v̄ as similarity score S_W(i, j)
  end for
  Sort S_W(1:N_s, j) in descending order /* sort images based on similarity score */
  Assign top k images to the set R_j for the class j
end for
Return R_j for each class j
```

Fig. 2. Recommendation algorithm---"CNN + Bilinear"

2.3 Data augmentation based on feature learning (Triplet + L2)

We train Facenet triplet learning model [8] on the seed training dataset T that aims at learning an embedding (feature representation) function $\mathcal{F}(I_i)$, from an image I_i into its corresponding feature vector by minimizing the overall loss L calculated using Equation 2. We want to achieve the goal that the squared distance between the image I_i and the image I_i^+ of the same class as I_i must be at least α smaller than the squared distance between the image I_i and image I_i^- of a different class as I_i as shown in Equation 3. The second term $\lambda \sum_{\theta \in P} \theta^2$ in Equation 2 is the regularization term [2] to prevent overfitting and obtain a smooth model. λ is the weight decay.

$$L = \sum_{i=1}^{N_\Gamma} \max(0, \|\mathcal{F}(I_i) - \mathcal{F}(I_i^+)\|_2^2 + \alpha - \|\mathcal{F}(I_i) - \mathcal{F}(I_i^-)\|_2^2) + \lambda \sum_{\theta \in P} \theta^2 \quad (2)$$

$$\|\mathcal{F}(I_i) - \mathcal{F}(I_i^+)\|_2^2 + \alpha < \|\mathcal{F}(I_i) - \mathcal{F}(I_i^-)\|_2^2, \ \forall (I_i, I_i^+, I_i^-) \in \Gamma \quad (3)$$

where α is an enforced margin between positive and negative pairs; P is the set of all parameters in $\mathcal{F}(I_i)$; I_i^+(positive) is an image from the same class as I_i. I_i^- (negative) is an image from a different class as I_i. Γ is the set of all possible triplets in the training set and has cardinality N_Γ. Fig. 3 shows our method based on the learned embedding function using the squared distance function (L2) to find similar images.

```
for j = 1, 2, ..., N_C
  Compute the center of the feature vectors of all images of
  the class j in T: v̄ = Σ_{i=1}^{N_j} F(I_i)/N_j
  for i = 1, 2, ..., N_S
    distance d(i, j) = ||F(I_i) − v̄||_2^2
  end for
  Sort d(1:N_S, j) in ascending order /* sort images based on L2 distance */
  Assign top k images to the set R_j for the class j
end for
Return R_j for each class j
```

Fig. 3. Recommendation algorithm---"Triplet + L2"

2.4 Unified Learning of Feature Representation and Similarity Matrix

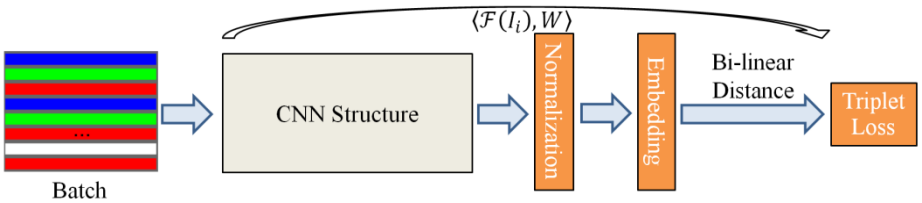


Fig. 4. The model consists of a batch input layer to a CNN followed by L2 normalization, which results in the embedding using the triplet loss based on the Bi-linear distance.

We describe our proposed **U**nified **L**earning of **F**eature Representation and **S**imilarity **M**atrix (Unified LF&SM). Fig. 4 shows the new model structure which is trained on the seed training dataset T. We aim at finding a similarity score model $S_{(\mathcal{F},W)}(I_i, I_j)$, which is a pair of an embedding function $\mathcal{F}(I_i)$ mapping an image I_i into a feature vector and a bilinear similarity matrix W, such that the similarity score between the image I_i and the image I_i^+ of the same class as I_i must be at least α bigger than the similarity score between the image I_i and image I_i^- of a different class as I_i as shown in Equations 4 and 5.

$$S_{(\mathcal{F},W)}(I_i, I_i^+) > S_{(\mathcal{F},W)}(I_i, I_i^-) + \alpha, \ \forall (I_i, I_i^+, I_i^-) \in \Gamma \tag{4}$$

$$S_{(\mathcal{F},W)}(I_i, I_j) = \left(\mathcal{F}(I_i)\right)^T W \mathcal{F}(I_j) \tag{5}$$

We minimize the loss function as shown in Equations 6 and 7 to obtain the above mentioned similarity score model.

$$L = \sum_{i=1}^{N_\Gamma} l_{\mathcal{F},W}(I_i, I_i^+, I_i^-) + \lambda \sum_{\theta \in P} \theta^2 \tag{6}$$

$$= \sum_{i=1}^{N_\Gamma} \max\left(0, \alpha - S_{(\mathcal{F},W)}(I_i, I_i^+) + S_{(\mathcal{F},W)}(I_i, I_i^-)\right) + \lambda \sum_{\theta \in P} \theta^2 \tag{7}$$

where the definition of α, N_Γ, I_i^- and I_i^+ are the same as in Section 2.3; P is the set of all parameters in $\mathcal{F}(I_i)$ and W. Unlike the Facenet model that uses L2 distance and optimizes for the feature representation, the new model does joint optimization on both the feature representation and the similarity learning function. Fig. 5 shows our recommendation algorithm using the learned similarity matrix and the learned feature representation to find unlabeled images similar to the training images for each class.

```
for j = 1, 2, ..., N_C
  Compute the center of the feature vectors of all images of
  the class j in T: v̄ = Σ_{i=1}^{N_j} F(I_i)/N_j
  for i = 1, 2, ..., N_S
     similarity score S_(F,W)(i, j) = F(I_i)^T W v̄
  end for
  Sort S_(F,W)(1:N_S, j) in descending order /* sort images based on similarity score */
  Assign top k images to the set R_j for the class j
end for
Return R_j for each class j
```

Fig. 5. Recommendation algorithm---"Unified LF&SM"

3 Experiments

To evaluate the performance of the four data augmentation methods, we selected two image classification problems in endoscopy video analysis: the instrument image detection [10] and the retroflexion image detection [11]. These two problems share a

Fig. 6. Sample images for the six classes. From left to right: left cable body, right cable body, forceps head, snare head, retroflexion, and no object.

Table 1. Average percentage of images belonging to each class calculated on the 25 training videos.

Class Name	Ratio (%)
Left cable body	2.01
Right cable body	3.82
Forceps head	2.04
Snare head	1.51
Retroflexion	0.80
No object	89.8

Table 2. Our CNN structure. The input and output sizes are described in rows × cols × #nodes. The kernel is specified as rows × cols × #filters, stride.

layer	Size-in	Size-out	kernel
Conv1	64x64x3	64x64x16	3x3x16,1
Pool1	64x64x16	32x32x16	2x2x16,2
Conv2	32x32x16	32x32x32	3x3x32,1
Pool2	32x32x32	16x16x32	2x2x32,2
Conv3	16x16x32	16x16x64	3x3x64,1
Pool3	16x16x64	8x8x64	2x2x64,2
Conv4	8x8x64	8x8x128	3x3x128,1
Pool4	8x8x128	4x4x128	2x2x128,2
Conv5	4x4x128	1x1x256	4x4x256,1

common unbalanced data problem; instrument images and retroflexion images are rare as the proportions of these images are very small as shown in Table 1. Fig. 6 shows sample images for left cable body, right cable body, forceps head, snare head, retroflexion, and no object class for common endoscopy images without any of the aforementioned objects. We solve these two problems using one six-class CNN classifier.

Training dataset: We extracted and labeled one frame for every five frames from 25 de-identified full-length endoscopic videos of colonoscopy and upper endoscopy captured using Fujinon or Olympus scopes. Finally, we get a training set of 9300 images (1400 training images and 150 validation images for each class, N_C = 6). Table 1 shows the average percentage of images belonging to each class calculated on the 25 training videos.

Unlabeled dataset $\mathcal{U}$ consists of 600,000 unlabeled images (N_s= 600,000) from 228 endoscopic videos by automatically extracting one frame for every ten frames. Each unlabeled video is different from any training video.

Test dataset consists of 21000 images (3500 test images for each class) from 58 endoscopic videos by automatically extracting one frame for every five frames. Each test video is different from any training video and unlabeled video. The test dataset contains many rare-class images with quite different appearances (e.g., different instrument colors or shapes) from the training images.

3.1 Model Parameters

Considering the fact that only a small training set is available, we use a CNN structure which is similar to the VGG Net [12], but has much fewer parameters, as shown in Table 2. Our CNN models accept RGB images with the size of 64*x*64 pixels. These images are from resizing the raw endoscopic images. We implemented our CNN models using Python and Google's TensorFlow library [13]. When training the CNN classifiers described in Sections 2.1 and 2.2, we set the batch size as 256 and the epoch number as 400. When training the CNN models described in Sections 2.3 and 2.4, we set the enforced margin α as 0.2, the weight decay λ as 0.001, the epoch number as 200 (400 batches per epoch, 6 classes per batch, and 512 images by random selection per class). We learned the bilinear similarity function in Section 2.2 using the Matlab code provided by the author of OASIS and set the iteration number as 10^8. The feature vector of each image comes from the output of the "Conv5" in Table 2. To show the advantage of the proposed real data augmentation over the traditional data augmentation, we used KERAS [14] to apply rotation (0^0~30^0), shearing (0~0.01), translation (0~0.01), zooming (0~0.01), and whitening on each image in the seed training dataset and synthesized 5600 images for each class to expand the seed traing dataset.

3.2 Performance Metrics and Comparison

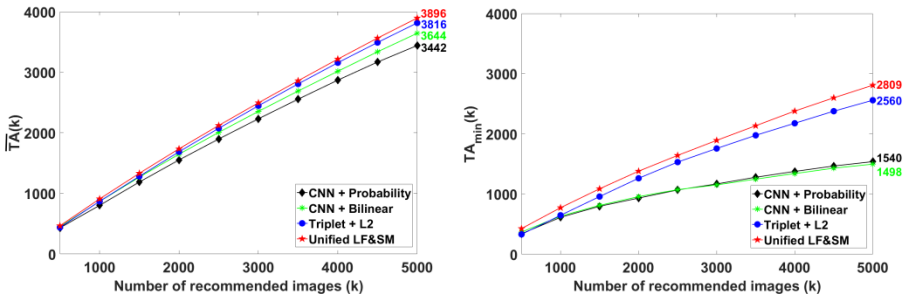
3.2.1 Classification Performance

Table 3. Comparison of 6-class image classification performance for different models.

Method	Average Recall	Average Precision
Baseline	80.3%	80.8%
Traditional Augmentation	83.2%	84.0%
CNN + Probability	84.4%	85.0%
CNN + Bilinear	85.8%	86.0%
Triplet + L2	88.9%	89.2%
Unified LF&SM	**89.3%**	**89.3%**

We trained the new CNN classifier by adding the new correctly recommended images (k=5000) to the seed dataset for each recommendation model and computed the average recall and average precision on the six classes. When training the CNN classifier for each method, we used the same CNN structure, weight decay, and learning rate.

In Table 3, Baseline represents the CNN classifier trained on the seed dataset. Table 3 shows that we can get the best average recall and average precision when using the Unified LF&SM to do real data augmentation. Table 3 also shows that, compared to the Baseline, the Unified LF&SM improved the average recall and the average precision by 9% and 8.5%, respectively. Table 3 also shows that even the simple method of selecting top k similar images still outperforms the traditional data augmentation that is commonly used. This result shows that our real data augmentation method is very useful for improving the image classification accuracy. Although the



(a) (b)

Fig. 7. (a) - (b) $\overline{TA}(k)$ and $TA_{min}(k)$ for the top k recommended images.

Table 4. Comparison of the number of images to be labeled using random selection and Unified LF&SM for each rare target class to obtain the same number of true accepts.

Class Name	#True Accepts	# Unified LF&SM	# Random Selection
Left cable	4600	5000	228860
Right cable	4992	5000	130680
Forceps head	2809	5000	137700
Snare head	3061	5000	202720
Retroflexion	2923	5000	365380

classification performance between Triplet+L2 and Unified LF&SM is very close, we will see next that Unified LF&SM reduces the efforts of manual labeling the most.

3.2.2 Efforts of Domain Experts

We define the number of true accepts (correct recommendations) in the top k recommended images for the class j as $TA(j,k)$. We define $\overline{TA}(k)$ as the average true accepts considering all classes for each k, a desired number of recommend images. We define $TA_{min}(k)$ as the number of true accepts for the class with the least correct recommendations among all the classes. We use the actual number instead of precision to reflect the medical experts' efforts to verify the recommended results.

$$\overline{TA}(k) = \sum_{j=1}^{N_c} TA(j,k)/N_c \qquad TA_{min}(k) = \min_{1\le j\le N_c} TA(j,k) \tag{8}$$

As shown in Fig. 7, the difference in true accepts increases as k increases. When k is small (<=1000), the difference in the correctness of the recommendation is small. As k becomes larger, the better technique makes more correct recommendations. Fig. 7 also shows that Unified LF&SM outperforms the three other methods by recommending 80~454 more true accepts (average number) and recommending 249~1311 more true accepts (minimum number) for the top 5000 recommendations. Although the difference on classification performance between Triplet + L2 training and Unified LF&SM in Table 3 is very small, but Unified LF&SM reduces the manual labeling workload as shown in Fig. 7. Fig. 7 (b) shows that when comparing the minimum

number of true accepts for all classes, Unified LF&SM and “Triplet + L2” show a much better result than “CNN + Bilinear” and “CNN + Probability.” The explanation is that the two latter methods have the class "snare head" as the class with the least correct recommendations and recommended fewer relevant images for the class “snare head”. One reason to explain the large performance difference is that the models using the triplet have many more training samples (N^3 in theory where N is the number of images in the training set) than those of the models using the single image input (only N) in the training process.

Assume we want to get k number of images belonging to the class j and the ratio of images belonging to the class j in the training video is r as shown in Table 1, then we estimate the number of images to be labeled using random selection as k/r in the fourth column of Table 4. For example, the estimated number is $202720 \approx 3061/(1.51\%)$ for the class “snare head”. Table 4 shows that, to obtain the same number of true accepts for a rare target class, medical experts have to verify at least 26 ($130680/5000 \approx 26$) times the number of images if using random selection of unlabeled images compared to if using Unified LF&SM. With Unified LF&SM, medical experts spend far less time on annotating ground truth, and still give adequate representative images for the rare target class.

3.3 Applicability to Other Types of Medical Images

Our Unified LF&SM automatically learns the image feature vector and the similarity matrix to recommend images when only given a small labelled image dataset. Therefore, the Unified LF&SM does not require specific domain knowledge on medical images and is easily extendable to other medical image classification problems.

4 Conclusion

We have presented and evaluated our Unified LF&SM with the goal to decrease the time needed for creating the training data by medical experts. We achieved this goal for the classification problems of instrument and retroflexion images. Our future work includes investigating a better recommendation algorithm, exploring active learning by repeatedly recommending images in iterations using the proposed Unified LF&SM, and extending the approach for object localization and temporal scene segmentation for medical image and video analysis.

References

1. Tajbakhsh, N., et al.: Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?. TMI **35**(5), 1299-1312 (2016)
2. Chatfield, K., et al.: Return of the Devil in the Details: Delving Deep into Convolutional Nets. arXiv preprint arXiv:1405.3531 (2014)
3. Shin, H.C., et al.: Learning to Read Chest X-Rays: Recurrent Neural Cascade Model for Automated Image Annotation. In: CVPR, pp. 2497-2506 (2016).

4. Zhu, X.: Semi-supervised Learning Literature Survey. 2005.
5. Lu, X., et al.: Enhancing Text Categorization with Semantic-enriched Representation and Training Data Augmentation. JAMIA **13**(5), 526-535 (2006)
6. Xu, Z., et al.: Augmenting Strong Supervision Using Web Data for Fine-grained Categorization. In: ICCV, pp. 2524-2532 (2015).
7. Chechik, G., et al.: Large Scale Online Learning of Image Similarity through Ranking, Journal of Machine Learning Research, **11**, 1109-1135 (2010)
8. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A Unified Embedding for Face Recognition and Clustering. In: CVPR, pp. 815-823 (2015)
9. Bishop, C.: Pattern Recognition and Machine Learning. Springer, pp. 144-146 (2007)
10. Zhang, C., et al.: Cable Footprint History: Spatio-Temporal Technique for Instrument Detection in Gastrointestinal Endoscopic Procedures. In: IPCV, pp. 308-314 (2015)
11. Wang, Y., et al.: Near Real-Time Retroflexion Detection in Colonoscopy. JBHI **17** (1), 143-152 (2013)
12. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-scale Image Recognition. arXiv preprint arXiv:1409.1556 (2014)
13. Abadi, M., et al.: TensorFlow: Large-scale Machine Learning on Heterogeneous Distributed Systems. arXiv preprint arXiv:1603.04467 (2016)
14. Chollet, F.: "Keras". https://github.com/fchollet/keras