Smart Mining for Deep Metric Learning

*Ben Harwood[2], *Vijay Kumar B G[1], Gustavo Carneiro[1], Ian Reid[1], Tom Drummond[2]
[1]The University of Adelaide,[2]Monash University
{vijay.kumar,gustavo.carneiro,ian.reid}@adelaide.edu.au, {ben.harwood,tom.drummond}@monash.edu

1. Effect of Parameters on the Embedding

In this section, we evaluate the performance of our proposed smart mining method with respect to various parameter settings. Note that in all our experiments (including the ones in the paper), we initialize the network with pre-trained GoogleNet [21] weights and randomly initialize the final fully connected layer similar to Lifted Structure [19]. We set the embedding size to 64 [19] and the learning rate for the randomly initialized fully connected layer is multiplied by 10 to achieve faster convergence similar to [19].

1.1. Effect of Scaling Parameter κ on the Embedding

We define smart triplets as those that satisfy Eq. 6, where κ is a global scaling factor that decides the radius of the hyper-spherical exclusion boundary centred around the anchor. In this sub-section, we show the effect of κ on the feature embedding. To this end, we run the experiments on CUB-200-2011 dataset for different initial values of $\kappa \in \{1, 4, 16, 64\}$. We use **Triplet + FANNG + Global** as the loss function and report the recall values at $1, 2, 4$ and 8 at the end of 20^{th} epoch. Fig. 1 shows that the performance degrades for smaller values of κ. This is due to hard triplets generated by the mining algorithm. For large values of κ, there are fewer smart triplets returned by the approximate nearest neighbor search, so random triplets are used instead. In the latter case, the behavior of the method tends to be similar to that of the **Triplet + Global**.

1.2. Effect of the Percentage of Mined Triplets for Training

Figure 2 shows the effect of varying the percentage of mined triplets for training on the CUB-200-2011 dataset. We train **Triplet + FANNG + Adaptive** networks for 20 epochs using a target training error of 0.5 and with the percentage of mined triplets varying from 10% to 60% in 10% increments. For these experiments the global loss has been disabled so that the training error is a result of only the triplet losses. At the lower percentages, there are insufficient mined triplets to properly control the training error and accelerate the training. From 40% mined triplet and beyond, there are enough mined triplets to allow for control the training error and so the performance begins to saturate at this level. As such, we find that a percentage of anywhere between 50% to 100% mined triplets is sufficient.

2. Visualizing Embedding using t-SNE

Fig. 3 shows the Barnes-Hut t-SNE visualisation of the learned embedding space obtained by mapping the CUB-200-2011 test image features to a two-dimensional space. Although, there is no overlap between the train and test classes, the images from the test classes are clustered well.

3. Sample Mined Triplets using FANNG

The images in Figure 4 are triplets from randomly selected anchor points while training **Triplet + FANNG + Adaptive** on the CUB-200-2011 dataset. Similar to the experiments in Section 1.2, we are interested in showing only the learnning resulting from the triplet mining and as such global loss is disabled. At epochs $4, 8, 12$ and 16 the first triplet formed for each of the chosen anchor points was recorded. Beginning with the epoch 4 images, visual inspection shows that the mined negative samples share distinct visual traits with the anchor image and hence they are already much harder than random

*Vijay Kumar B G and Ben Harwood contributed equally to this work

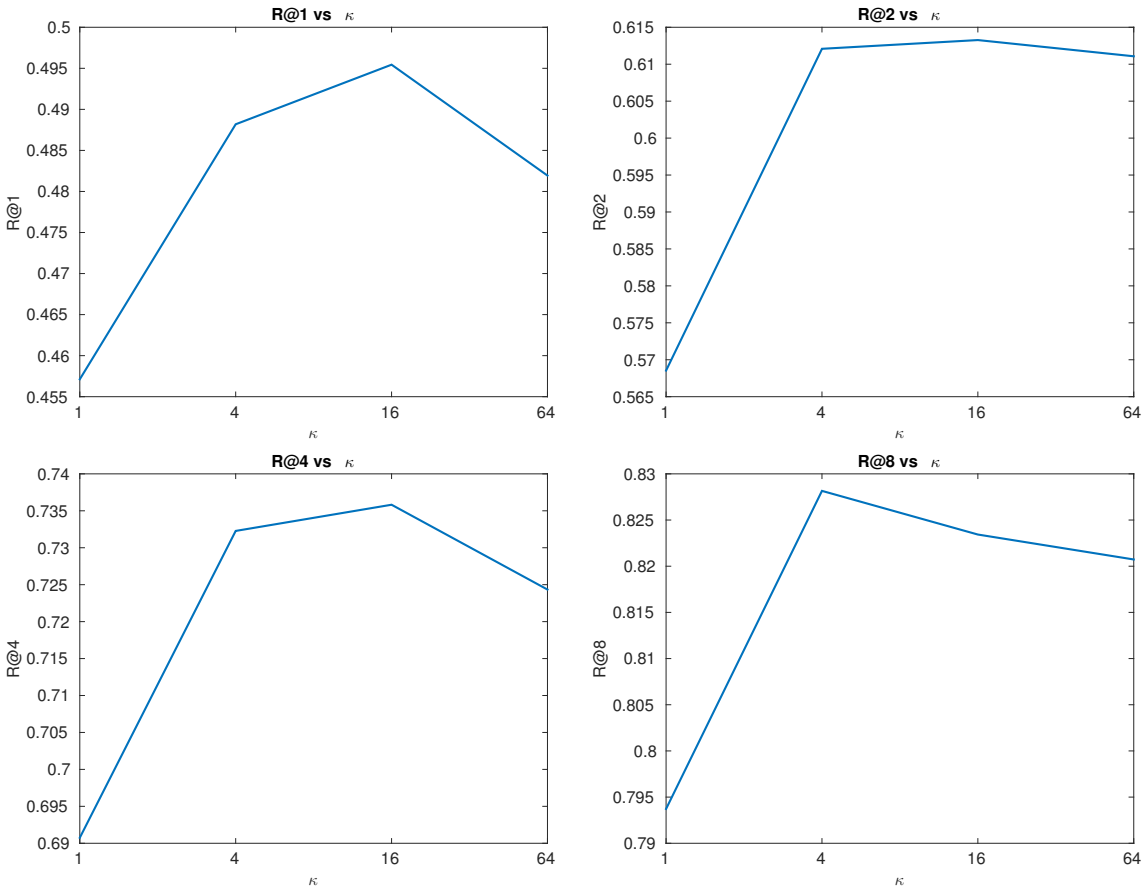

Figure 1. $R@1$ vs κ (top left), $R@2$ vs κ (top right), $R@4$vs κ (bottom left), $R@8$ vs κ (bottom right)

negatives. Beyond epoch 4, the mined negatives continue to become more difficult as the embedding is refined. In particular, many of the negative images at Epoch 16 could easily be mistaken as coming from the same class as the anchor image. The appearence of the positive samples is largly constrained by the negatives, since our method always selects the softest positive that is also still harder that the chosen negative. This selection process can be seen in the way each positive-negative pair share many distinctive visual traits such that they are roughly the same distance from the anchor point. However, in some cases the negative and positive samples could be in very different directions from the anchor, and so visually judging the similar level of difficulty is much more difficult across different regions of the embedding.

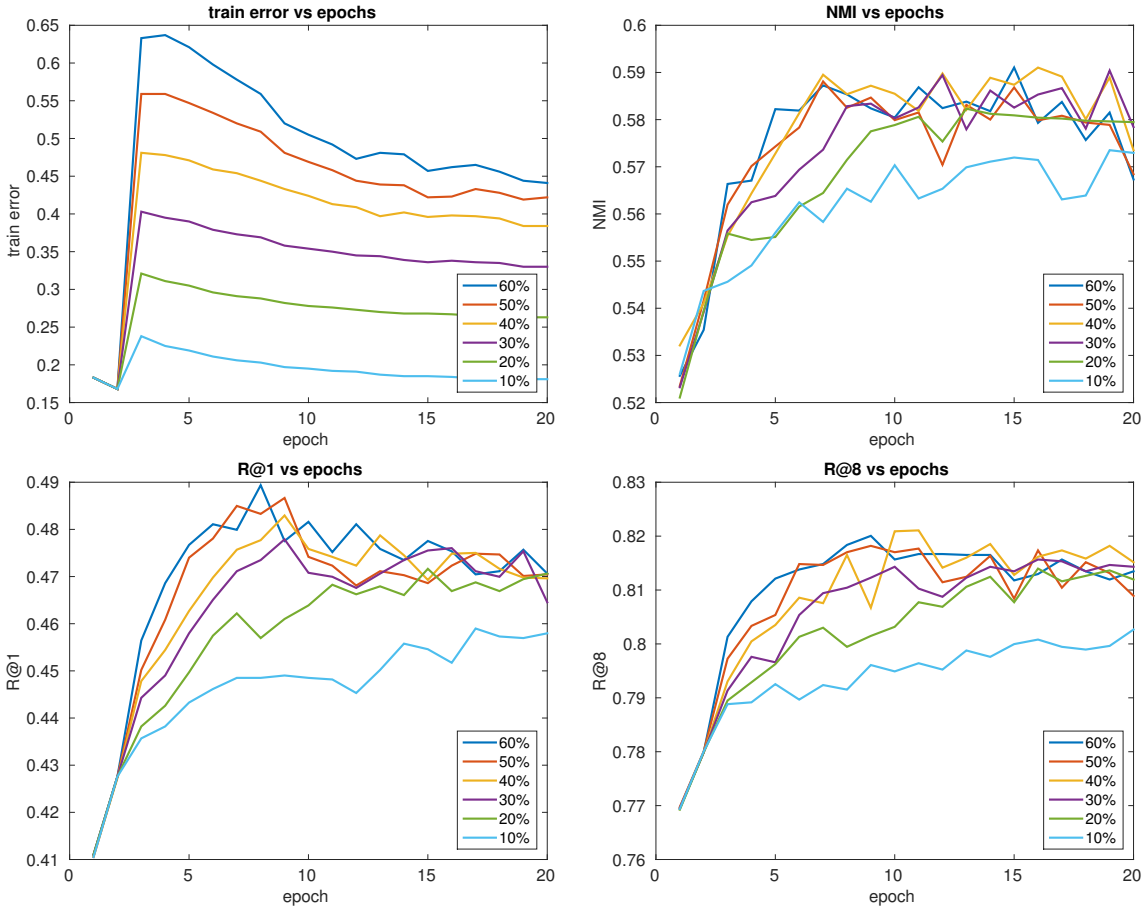

Figure 2. Training error vs epoch (top left), NMI vs epoch (top right), $R@1$vs epoch (bottom left), $R@8$ vs epoch (bottom right)

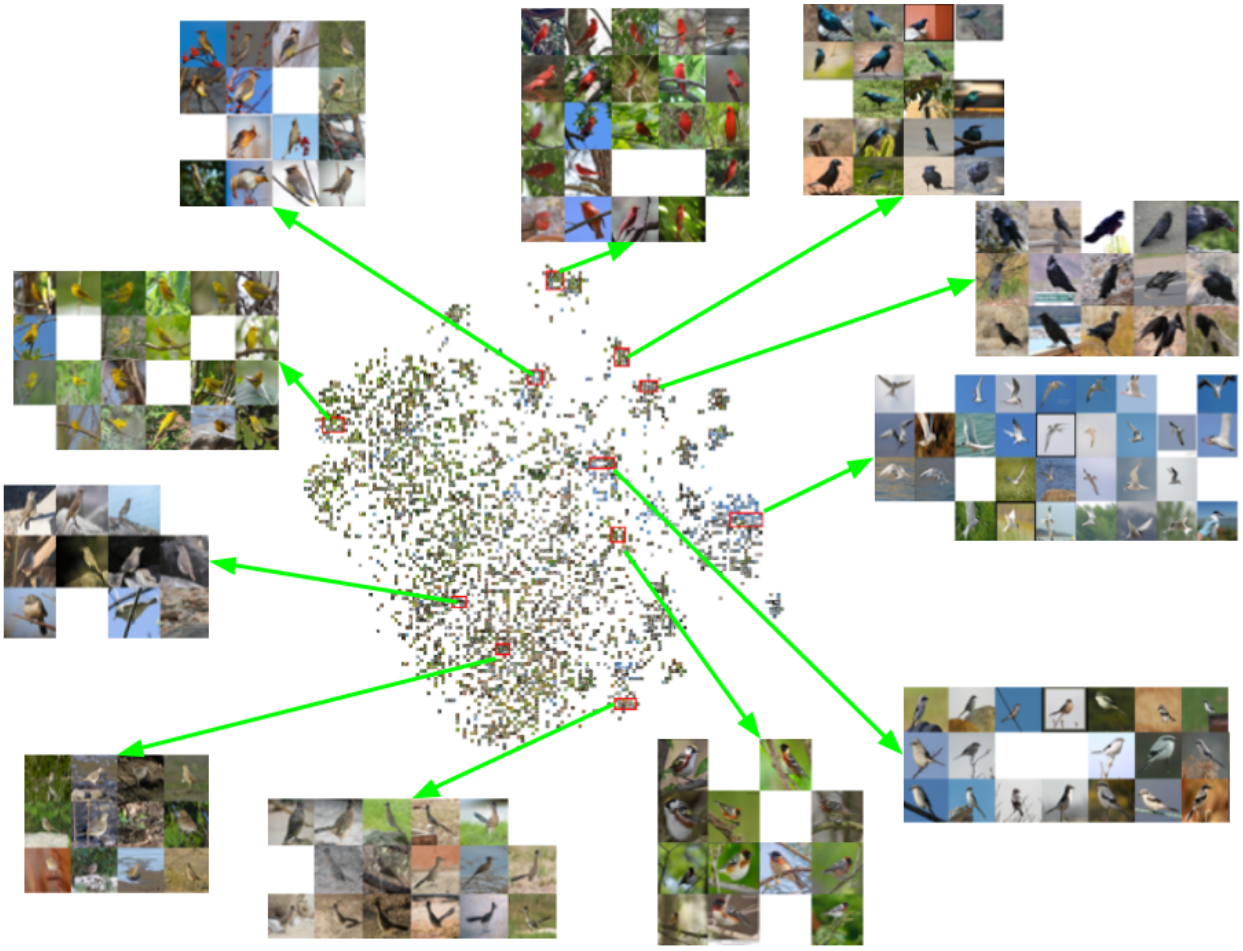

Figure 3. Barnes-Hut t-SNE visualization of the CUB-200-2011 test images

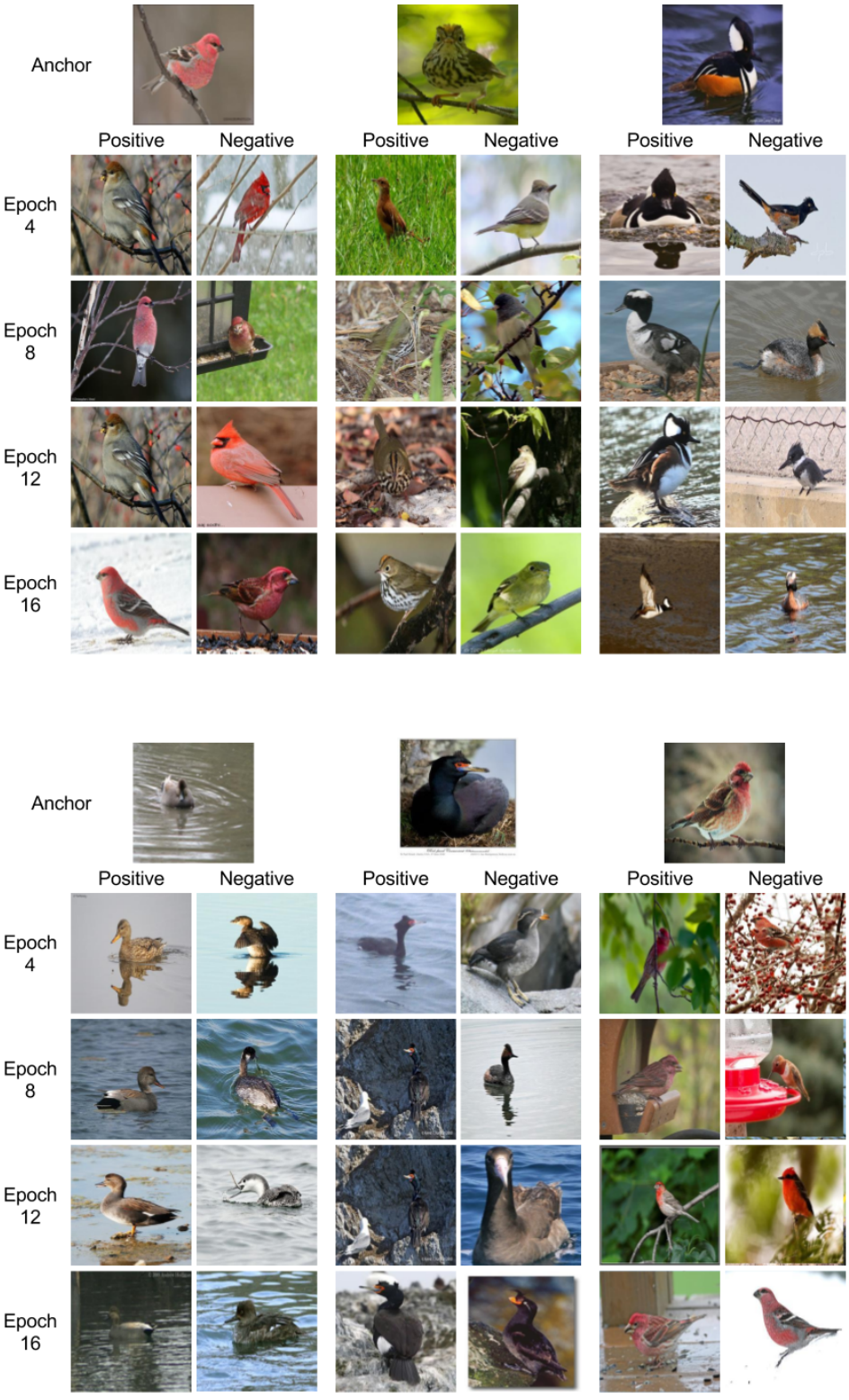

Figure 4. Mined triplets for 6 specific anchor points at training epochs $4, 8, 12$ and 16.