DeephESC: An Automated System for Generating and Classification of Human Embryonic Stem Cells

Rajkumar Theagarajan, Benjamin X. Guan and Bir Bhanu
Center for Research in Intelligent Systems
University of California, Riverside, Riverside, CA - 92521
Email: {rthea001, xguan001, bhanu.ee}@ucr.edu

***Abstract*—Human Embryonic Stem Cells (hESC's) are promising for the treatment of many diseases such as cancer, Parkinsons, Huntingtons, diabetes mellitus etc. and for toxicological testing. Automated detection and classification of human embryonic stem cell (hESC) videos is of great interest among biologists for quantified analysis of various states of hESC in experimental work. To date, the biologists who study hESC's have to analyze stem cell videos manually. In this paper we introduce a hierarchical classification system consisting of Convolutional Neural Networks (CNN) and Triplet CNN's to classify hESC images into six different classes. We also design an ensemble of Generative Adversarial Networks (GAN) for generating synthetic images of hESC's. We validate the quality of the generated hESC images by training all of our CNN's exclusively on the synthetic images generated by the GAN's and evaluating them on the original hESC images. Experimental results shows that we classify the original hESC images, with an accuracy of 85.67% using the CNN alone, 91.38% accuracy using the CNN and Triplet CNN and 94.11% accuracy by fusing the outputs of the CNN and Triplet CNN's, out performing existing state-of-the-art approaches.**

I. Introduction

Video Bioinformatics [21] is an upcoming field to help biologists achieve quicker and less demanding approaches to analyze expansive volumes of video data. The biologists who study human embryonic stem cells (hESC's) have to analyze stem cell videos every day. This is a very laborious manual process and the accuracy of a human performing classification is inversely proportional to long working hours. It is important to utilize the information from live videos to study the behavior of hESC's during exposure to various chemical agents.

hESC's are derived from the inner cell mass of developing blastocysts and possess two important properties: 1) self-renewal and 2) pluripotency [1]- [3]. Self-renewal is the ability to go through unlimited cycles of cell division, and pluripotency is the capability to differentiate into any cell type in the human body. hESC are an important resource for regenerative medicine, basic research on human prenatal development, and toxicological testing of drugs and environmental chemicals. Under the state of pluripotency, they can also be maintained indefinitely [4] [5]. Therefore, understanding the behavior of hESC is fundamental for medicinal and toxicological research [5] - [7].

In this paper, we propose a system for classifying hESC images using Convolutional Neural Networks (CNN) and Triplet CNN's in a hierarchical system. We compensate for the lack of data by training Deep Convolutional Generative Adversarial Networks (DCGAN) to generate synthetic hESC images. We also make use of the features learned by individual GAN's, to further improve the quality of the generated images. Furthermore, we validate the quality of the generated synthetic images by training the CNN and Triplet CNN's exclusively on the synthetic images and evaluated the networks on the original hESC images.

In this study, the hESC videos were taken with a BioStation IM [8]. The hESC videos consist of frames of phase contrast images. Each frame can contain any of the following six cell types: 1) Cell Clusters ***(CC)***, 2) Debris ***(DEB)***, 3) Unattached Cells ***(UN)***, 4) Attached Cells ***(AT)***, 5) Dynamically Blebbing Cells ***(DYN)***, and 6) Apoptotically Blebbing cells ***(APO)***. Fig. 1 shows phase contrast images for each class.

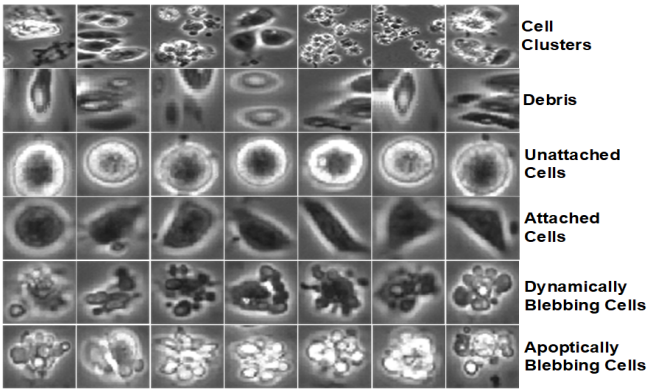

Fig. 1. Phase contrast images for the six different classes.

The unattached cells, attached cells, dynamically blebbing cells and apoptically blebbing cells are considered as the intrinsic cell types. Cell Clusters are a colony of cells consisting of different intrinsic cell types that are packed close to each other. Blebbing cells are membrane protrusions that appear and disappear from the surface of cells. The changing area of the blebbing cells over time is important for understanding and evaluating the health of cells. Dynamic blebs indicate healthy cells and apoptotic blebs indicate dying cells. The ability to analyze rates of bleb formation and retraction are important in the field of toxicology and could form the basis of an assay that depends on a functional cytoskeleton [4] [6].

II. Related Works and Our Contributions

There has been very limited work for stem cell classification in the field of computer vision. Guan *et al*. [3] [10] [12] proposed a model based approach for automatically segmenting hESC's. The authors considered the foreground and background intensity distribution as a mixture of two Gaussians. The objective of the algorithm is to find a threshold that optimizes a criterion derived from the intensity distribution of the foreground and background. The optimal segmentation is achieved at the highest criterion value. Since the segmentation method yields a binary image for each frame, the authors were able to extract a pool of individual components from each frame. The authors reported that, their proposed method yields less than 10% average detection error of foreground and background.

Lowri *et al*. [11] designed a texture based multi-stage Bayesian level set algorithm to segment colony images of stem cells and their derivatives. The authors use a Gabor based filter bank to segment multi resolution texture images belonging to different derivatives of stem cells.

Mangoubi *et al*. [22] classified hESC into differentiated and pluripotent cell colonies using a wavelet based texture decomposition. The authors observed that in contrast to differentiated colonies, pluripotent colonies contained homogeneous tight textures, thus allowing a statistical analysis of the coefficients obtained from a wavelet based texture decomposition to discriminate between the colonies. The authors achieved an accuracy of 96% in classifying colonies that were very distinct from each other and 86% in colonies with a mixed distribution.

To the best of our knowledge, we are not aware of any other approach that uses Convolutional Neural Networks (CNN) to classify hESC images into six different classes.

In light of the state-of-the-art, the contributions of this paper are:

- A hierarchical classification system using CNN and Triplet CNN's to classify hESC images.
- Training an ensemble of Generative Adversarial Networks (GAN) and used the features learned by individual GAN's to improve the quality of generated images.
- Validated the quality of the generated images by training all of our networks on the generated hESC images and evaluating them on the original hESC images. We further compare our approach with state-of-the-art classifiers.

III. Technical Approach

In this section we explain the framework and architecture of our approach, the individual models in our hierarchical classifier and the Generative Adversarial Networks used for generating synthetic images. Fig. 2 shows the overall architecture of our hierarchical classification.

A. Detection of hESC Regions

In our system, we use the approach proposed by Guan *et al*. [3] for detecting the cell regions. The hESC's are grown in culture dishes coated with a layer of substrate (Matrigel). The substrate becomes the background after the hESC's are placed on its surface. Therefore, we model a hESC image with two regions of interest: foreground and background. Fig. 1 shows that the intensity distributions of these regions are similar to a mixture of two Gaussians with different means and variances.

The objective of the algorithm is to find an optimal threshold that maximizes the absolute difference of the two mean-to-variance ratios *M* derived from the intensity distribution of foreground and background.

$$M = \left| \frac{\mu_f}{\sigma_f} - \frac{\mu_b}{\sigma_b} \right| \tag{1}$$

The optimal segmentation is achieved at the highest criterion value. The segmentation algorithm was evaluated using the Intersection over Union (IoU) metric and achieved an accuracy of **86.6%**. After detecting each individual cell in a given frame, they are classified into the six different classes using our hierarchical classification system.

B. Hierarchical Classification of hESC Images

Recently CNN's and Triplet CNN's have shown to be very robust in extracting features for biological applications such as detection of different types of cancer, gene identification and biometrics liveness detection [13] - [15]. In our approach we use CNN's and Triplet CNN's in a hierarchical manner to classify hESC images into the six different classes.

1) Convolutional Neural Network: After detecting all the cell regions in the video, we resize all the hESC images to size 64x64. These images are then used for training the CNN. Table I. shows the architecture of our CNN.

In Table I, conv(x, y, z) represents convolution(kernel size=x, stride=y, padding=z) similarly, maxpool(x, y, z) represents maxpool(kernel size=x, stride=y, padding=z). The weights for the CNN were initialized with uniform Xavier distribution as described in [16].

Input dimension	Output dimension	Number of Feature maps	Layer
64x64	32x32	64	Conv(7,2,3)
32x32	16x16	64	Maxpool(3,2,1)
16x16	8x8	128	Conv(5,2,2)
8x8	4x4	128	Maxpool(3,2,1)
2,048x1	6 classes	-	FC layer

TABLE I
ARCHITECTURE OF CNN

To train the CNN, we chose a mini batch size of 64. Since the size of our dataset is very limited, in order to prevent the CNN from over-fitting, we perform random affine transformations to the images. The images are randomly rotated between $-180°$ to $180°$, randomly sheared between $0°$ to $30°$, randomly zoomed in and out between 70% to 140% of the image size and randomly horizontally flipped. Furthermore, we randomly chose 10 images from each class as the validation dataset. The remaining of the dataset was divided into 2 folds, 50% of the dataset for training and 50% for testing

We did random hyper-parameter search for the CNN to obtain the best learning rate, momentum and weight decay.

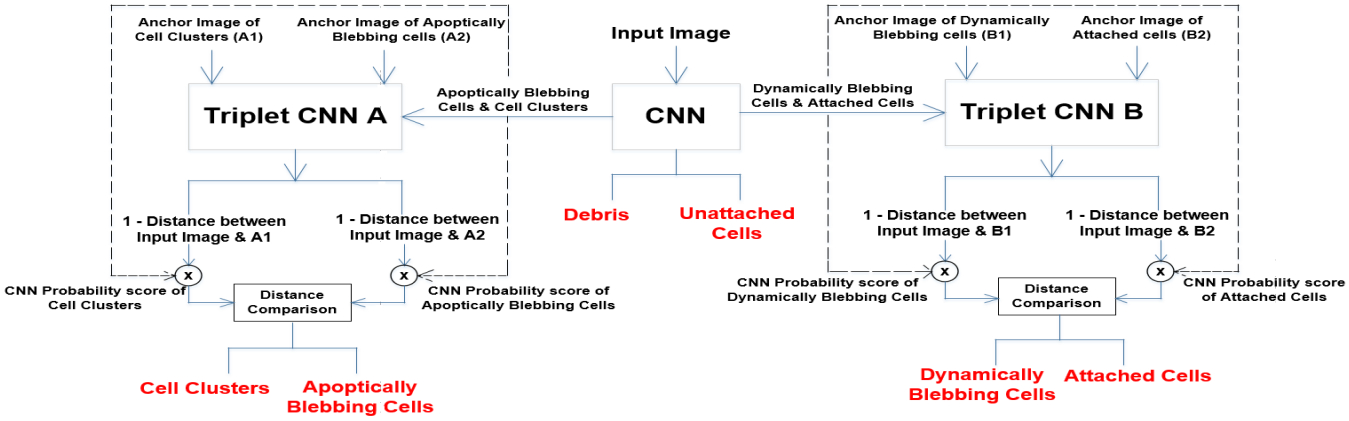

Fig. 2. Overall architecture of our hierarchical classification.

We chose random values for the learning rate, momentum and weight decay within a given range and step size and trained the network for three epochs. The combination of hyper-parameters that gave us the highest classification accuracy after three epochs are chosen as the best hyper-parameters for the network. The random hyper-parameter search was done by evaluating the CNN only on the validation dataset. Based on this we chose the best hyper-parameters as learning rate = $1.2\text{x}10^{-2}$, momentum = 0.9 and weight decay = $1\text{x}10^{-3}$ The network was optimized using the stochastic gradient descent algorithm with cross entropy loss.

We performed 2-fold cross validation and the results are shown in detail in the experimental section. After evaluating the CNN we observed that the CNN was able to classify the classes Debris and Unattached Cells with high accuracy, but the classes Cell Clusters/ Apoptically Blebbing cells and Dynamically Blebbing Cells/ Attached Cells were misclassified the most. The reason for this is that, the classes Cell Clusters/ Apoptically Blebbing Cells and Dynamically Blebbing Cells/ Attached Cells have similar intensity and texture. The only difference between these classes is their morphology.

2) Triplet Convolutional Neural Networks: To solve this misclassification, we train a Triplet CNN to perform fine-grained classification between Cell Clusters and Apoptically Blebbing Cells and similarly, for Dynamically Blebbing Cells and Attached Cells. Table II. shows the architecture for Triplet CNN A and Triplet CNN B shown in Fig. 2.

Input dimension	Output dimension	Number of Feature maps	Layer
64x64	32x32	256	Conv(7,2,3)
32x32	16x16	256	Maxpool(3,2,1)
16x16	8x8	512	Conv(5,2,2)
8x8	4x4	512	Conv(3,2,1)
8,192x1	2 classes	-	FC layer

TABLE II
ARCHITECTURE OF TRIPLET CNN A AND TRIPLET CNN B

The Triplet CNN takes as input a query image and one anchor image from each class. The output of the Triplet CNN is the two pairwise distances between the extracted features for the query image and the two anchor images as shown in Fig. 2. The query image is classified as the same class as the anchor image closest to the query image.

We used the same 10 validation images from each class used for validating the CNN, to validate the Triplet CNN. To train both Triplet CNN A and Triplet CNN B, we did 2-fold cross validation similar to how we trained the CNN. From each fold, we randomly selected 75,000 triplets for training, 3,000 triplets for validation and 35,000 triplets for testing. We chose a mini-batch size of 256 triplets and performed random affine transformation to the images and random hyper-parameter search that was similarly done while training the CNN. Table III. shows a summary for the best hyper-parameters for the CNN, Triplet CNN A and Triplet CNN B. The Triplet CNN's were optimized using the Stochastic Gradient Descent algorithm with the Ranked Marginal loss function as shown in equation 2.

$$Loss = Max(0, -Y * (G(X_1) - G(X_2)) + margin) \tag{2}$$

Network	Learning rate	Momentum	Weight decay
CNN	$1.2\text{x}10^{-2}$	0.9	$1\text{x}10^{-3}$
Triplet CNN A	$5\text{x}10^{-2}$	0.75	$1\text{x}10^{-4}$
Triplet CNN B	$6\text{x}10^{-2}$	0.8	$1\text{x}10^{-4}$

TABLE III
BEST HYPER-PARAMETERS FOR TRAINING THE NETWORKS

In equation 2, *G(X)* is the pairwise distance between the 1D feature extracted by fully connected layer of the Triplet CNN for the query image and the anchor image, *Y* is the ground-truth. If *Y* = 1, it is assumed the first input should be ranked higher than the second input, and vice-versa for *Y* = -1.

Upon evaluating the Triplet CNN's with 2-folds cross validation, Triplet CNN A achieved an average classification accuracy of 91.86% and Triplet CNN B achieved an average classification accuracy of 95.73%.

After training the CNN and the individual Triplet CNN's we combine them in a hierarchical system as shown in Fig. 2. The input hESC image is first passed into the CNN, the CNN classifies the input image into one of the six classes. If the predicted class is Debris or Unattached cells, we take the prediction of the CNN as final prediction.

If the predicted class is Cell Clusters or Apoptically Blebbing cells, the input image is passed to Triplet CNN A, and we take the prediction of Triplet CNN A as final prediction. Similarly, if the prediction of the CNN is Attached cells or Dynamically Blebbing cells, the input image is passed to the Triplet CNN B and we take the prediction of Triplet CNN B as final prediction. The results obtained are explained in detail in the experimental section.

3) Fusing the Output of CNN and Triplet CNN: In an attempt to further improve the classification accuracy, we fused the outputs from both the CNN and Triplet CNN's. The fusion was done by taking the complimentary pairwise distance measure outputs from the Triplet CNN and multiplying the corresponding probability score for that class from the CNN. For example in Fig. 2, in Triplet CNN A, the complimentary pairwise distance measure between the input image and anchor image of Cell Clusters is multiplied with the probability score for Cell Clusters from the CNN. Similarly, the complimentary pairwise distance measure between the input image and anchor image of Apoptically Blebbing cells is multiplied with the probability score for Apoptically Blebbing cells from the CNN, and so on for Triplet CNN B. The results obtained are explained in the experimental section.

C. Generating hESC Images Using Ensemble of GAN's

In this section we explain on how we performed data augmentation to our dataset. The purpose of data augmentation is to determine if adding more variability to the training dataset helps improve the performance of the network.To achieve this we trained a state-of-the-art Deep Convolutional Generative Adversarial Network (DCGAN) [17].

DCGAN consists of two deep convolutional neural networks, a generator G and a discriminator D trained against each other. The generator takes a random noise vector z and returns an image $X_{gen} = G(z)$. The discriminator takes a real or a generated image, and outputs a probability distribution $P(S|X) = D(X)$ over the two image sources S. The discriminator is trained to maximize the log-likelihood of assigning the correct source while G tries to minimize it:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \\ +\mathbb{E}_{x\sim p_z(z)}[\log(1-D(G(z)))] \quad (3)$$

The objective is that the two networks converge to the Nash equilibrium so that D is maximally confused and G generates samples that resemble the training data.

In our approach we trained six individual DCGAN's to generate images belonging to the corresponding six classes. Table IV and Table V shows the architecture of the discriminator and generator, respectively.

Input dimension	Output dimension	Number of Feature maps	Layer
64x64	32x32	32	Conv(7,2,3)
32x32	16x16	64	Conv(3,2,1)
16x16	8x8	128	Conv(5,2,2)
8x8	4x4	256	Conv(3,2,1)
4,096x1	1 class	-	FC layer

TABLE IV
ARCHITECTURE OF THE DISCRIMINATOR

Input dimension	Output dimension	Number of Feature maps	Layer
100x1	8,192x1	-	FC layer
4x4	8x8	512	ConvT(6,2,2)
8x8	16x16	256	ConvT(6,2,2)
16x16	32x32	128	ConvT(6,2,2)
32x32	64x64	1	ConvT(6,2,2)

TABLE V
ARCHITECTURE OF THE GENERATOR

In Table V, convT(x, y, z) represents convolution transpose(kernel size=x, stride=y, padding=z). We chose the learning rate for the Generator to be $1x10^{-4}$ and learning rate of the Discriminator to be $1x10^{-5}$ and mini batch of size 32. Both the networks were optimized using the Adam algorithm with loss function as a combination of Binary Cross Entropy and Embedding loss as shown in equation 4.

$$Loss = \frac{-1}{n}\sum_{i=1}^{n} y_i * log(p_i) + (1-y_i) * log(1-p_i) \\ +\alpha * \frac{1}{n}\sum_{i=1}^{n} ||X_i - X||^2 \quad (4)$$

In equation 4, the first term is the Binary Cross Entropy loss. y_i is the ground-truth label (real or fake image), p_i is the probability score being a real image. The second term is the Embedding loss, X_i is an image from the mini batch (either fake or real image) and X is a real image chosen randomly from the training dataset.

The Binary Cross Entropy loss ensures that the GAN is able to extract accurate features to generate synthetic images resembling the images from the training dataset and the Embedding loss ensures that the generated images have same morphology as the images from the training dataset. α is an empirical value and was chosen to be $5x10^{-2}$. Fig. 3 shows examples of generated image using the individual GAN's.

On comparing the generated images in Fig. 3 with original images in Fig. 1, we found that the images belonging to the class Cell Clusters were not of good quality and had few artifacts. By definition, Cell Clusters are a colony of cells consisting of the four intrinsic cell types (unattached, attached, dynamically blebbing and apoptically blebbing cells) tightly packed in a very small area.

In order to generate good quality images of Cell Clusters, we used the features learned by the GAN's corresponding to the intrinsic cells namely: Attached cells, Unattached cells, Dynamically Blebbing Cells and Apoptically Blebbing cells

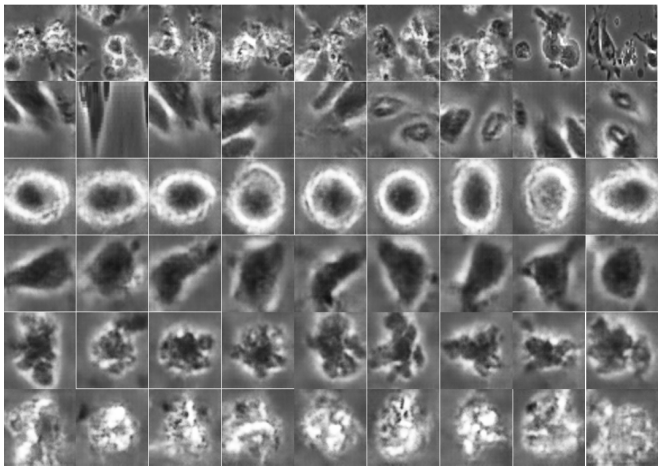

Fig. 3. Generated images for the six different classes. Each row corresponds to the class of the real images as shown in Fig. 1.

in an ensemble manner to train another GAN to generate Cell Clusters.

This is achieved by giving the same random noise vector Z as input to all the four Generators. The generated images are passed to the corresponding Discriminators. The features extracted from the fully connected layer of the corresponding discriminators are concatenated together and given as input to train a GAN to generate Cell Clusters. The input vector to this generator is a 1D feature vector of dimension 16,384x1. The Discriminator for generating Cell Clusters is the same as in Table IV, and Table VI shows the architecture of the Generator. Fig. 4 shows examples of generated Cell Cluster images using our ensemble of GAN's approach.

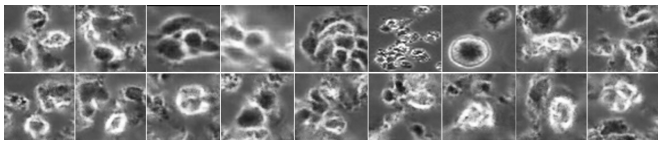

Fig. 4. Generated images for Cell Clusters using ensemble of GAN's.

Input dimension	Output dimension	Number of Feature maps	Layer
16,384x1	4,096x1	-	FC layer
4,096x1	8,192x1	-	FC layer
4x4	8x8	512	ConvT(6,2,2)
8x8	16x16	256	ConvT(6,2,2)
16x16	32x32	128	ConvT(6,2,2)
32x32	64x64	1	ConvT(6,2,2)

TABLE VI
ARCHITECTURE OF THE GENERATOR FOR CELL CLUSTERS

We further validated the quality of the generated images, by training the CNN and Triplet CNN's exclusively on the synthetic hESC images and evaluated them on the real hESC images. The results are shown in the experimental section.

IV. EXPERIMENTAL RESULTS

A. Dataset

The videos were acquired using a 20x objective with 600x800 resolution. A dataset of 784 images was obtained from nine hESC videos. The dataset had the following numbers of images for each class: 1) 122 Cell Cluster images; 2) 113 Debris images; 3) 135 Unattached cell images; 4) 132 Attached cell images; 5) 104 Dynamically Blebbing cell images; and 6) 178 Apoptotically Blebbing cell images. The ground-truth for the dataset was generated manually by expert stem cell biologists.

B. Classification Results

Table VII shows the average classification accuracy for the 2-fold cross validation using CNN, CNN-Triplets, Fused CNN-Triplets. All the networks in Table VII were trained and evaluated on the real hESC images.

Approach	Average Classification Accuracy
CNN	86.14%
CNN-Triplet	89.37%
Fused CNN-Triplet	**91.71%**

TABLE VII
AVERAGE CLASSIFICATION ACCURACY USING REAL HESC IMAGES

Table VIII and Table IX shows the confusion matrix using the CNN and Fused CNN-Triplet for classification. It can be observed in Table VIII, Cell Clusters **(CC)**/Apoptically Blebbing cells **(APO)** and Attached cells **(AT)**/Dynamically Blebbing cells **(DYN)** were misclassified the most. It can be observed in Table IX, that our approach of using a Fused CNN-Triplet hierarchical classifier helps resolve the misclassification and improve the overall accuracy.

Class	CC	DEB	UN	AT	DYN	APO
CC	**95**	2	0	1	1	13
DEB	1	**100**	0	0	2	0
UN	3	0	**121**	0	0	1
AT	1	2	0	**97**	20	2
DYN	2	0	0	12	**80**	0
APO	27	3	2	0	5	**131**

TABLE VIII
CONFUSION MATRIX USING CNN FOR CLASSIFICATION

Class	CC	DEB	UN	AT	DYN	APO
CC	**102**	2	0	1	1	6
DEB	1	**100**	0	0	2	0
UN	3	0	**121**	0	0	1
AT	1	2	0	**108**	9	2
DYN	2	0	0	2	**90**	0
APO	15	3	2	0	5	**143**

TABLE IX
CONFUSION MATRIX USING FUSED CNN-TRIPLET FOR CLASSIFICATION

In order to observe the effect of augmenting the dataset with the generated synthetic hESC images, we generated 30,000 images for each class using our ensemble of GAN's. Table X, shows the classification accuracy using only the generated images for training. We also compared our approach with existing

state-of-the-art CNN classifiers. All the networks in Table X were trained exclusively only on the generated synthetic hESC images and evaluated on the real hESC images.

Approach	Classification Accuracy
CNN	85.67%
CNN-Triplet	91.38%
Fused CNN-Triplet	**94.07**%
ResNet18 [18]	67.31%
ResNet34 [18]	66.72%
ResNet50 [18]	70.87%
VGG-16 [19]	77.34%
VGG-19 [19]	79.35%
AlexNet [20]	73.47%

TABLE X
COMPARISON OF CLASSIFICATION ACCURACY

It can be observed from Table X, the state-of-the-art approaches did not perform well. One of the reasons for this is that, these networks take as input, images of size 224x224, resizing the input images to this size causes a lot of pixel distortion making the images unrecognizable even to the stem cell biologists. Fig. 5 shows some of the misclassified images from our approach in Table X.

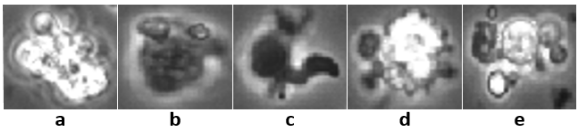

Fig. 5. Misclassified images from the Fused CNN-Triplet approach.

In Fig. 5, 5(a) is misclassified as Apoptically Blebbing cell, 5(b) is misclassified as Dynamically Blebbing cell, 5(c) is misclassified as Attached cell and 5(d) and 5(e) are misclassified as classified as Cell Clusters.

V. CONCLUSIONS

We proposed an automated system for detecting and classifying hESC images. We observed that the classes Cell Clusters/Apoptically Blebbing cells and Attached cells/Dynamically Blebbing cells have similar texture and intensity and were misclassified the most. We solved this problem by performing fine-grained classification using a hierarchical classification approach involving CNN and Triplet CNN's. We show that by fusing the outputs of the CNN and Triplet CNN's we can achieve higher classification accuracy.

Furthermore, we designed individual GAN's for each class to generate synthetic hESC images. We further improve the quality of the generated images by collectively using the features learned by individual GAN's in an ensemble. We validated the quality of the generated images, by training the entire system exclusively on the synthetic hESC images and evaluated them on the real hESC images.

VI. ACKNOWLEDGEMENT

This work was supported in part by NSF grant 1552454 and ONR grant N00014-12-1-1026 . The contents of the information do not reflect the position or policy of US Government.

REFERENCES

[1] J. Nichols and A. Smith, "The original and identity of embryonic stem cells," *Development*, vol. 138, pp. 3-8, January 2011.

[2] J.A. Thomson, J. Itskovitz-Eldor, S.S. Shapiro, M.A. Waknitz, J.J. Swiergiel, V.S. Marshall and J.M. Jones, "Embryonic stem cell lines derived from human blastocysts," *Science*, vol. 282, no. 5395, pp. 1145-1147, November 1998.

[3] B. X. Guan, B. Bhanu, P. Talbot and S. Lin, "Bio-driven cell region detection in human embryonic stem cell assay," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 3, pp. 604-611, 2014.

[4] Z. Zhu and D. Huangfu, "Human pluripotent stem cells: an emerging model in developmental biology," *Development*, vol. 140, pp. 705-717, 2013.

[5] P. Talbot and S. Lin, "Mouse and human embryonic stem cells: Can they improve human health by preventing disease?," *Current Topics in Medicinal Chemistry*, vol. 11, no. 13, pp. 1638-1652, 2011.

[6] S. Lin, S. Fonteno, J.H. Weng and P. Talbot, "Comparison of the toxicity of smoke from conventional and harm reduction cigarettes using human embryonic stem cells," *Toxicology Science*, vol. 118, pp. 202-212, August 2010.

[7] P. Talbot, N. zur Nieden, S. Lin, I. Martinez, B.X. Guan and B. Bhanu, "Use of video bioinformatic tools in stem cell toxicology," *Handbook on Nanotoxicology, Nanomedicine and Stem Cell Use in Toxicology*, April 2013.

[8] Nikon. Biostation-IM. http://www.nikoninstruments.com/Products/Cell-IncubatorObservation/BioStation-IM.

[9] S. Lin, S. Fonteno, J.H. Weng and P. Talbot, "Comparison of the toxicity of smoke from conventional and harm reduction cigarettes using human embryonic syem cells", *Toxicology Science*, vol. 118, pp. 202212, Aug. 2010

[10] B.X. Guan, B. Bhanu, N.S. Thakoor, P. Talbot and S. Lin, S, (2011, July). "Human embryonic stem cell detection by spatial information and mixture of Gaussians," *IEEE Healthcare Informatics, Imaging and Systems Biology*, pp. 307-314, July 2011.

[11] N. Lowry, R. Mangoubi, M. Desai, Y. Marzouk and P. Sammak, "Texton-based segmentation and classification of human embryonic stem cell colonies using multi-stage Bayesian level sets", *IEEE International Symposium on Biomedical Imaging*, pp. 194-197, 2012.

[12] B.X. Guan, B. Bhanu, P. Talbot and S. Lin, "Detection of non-dynamic blebbing single unattached human embryonic stem cells", *IEEE International Conference on Image Processing*, pp. 2293-2296, September 2012.

[13] P. Danaee, R. Ghaeini, and D.A. Hendrix, "A deep learning approach for cancer detection and relevant gene identification", *In Pacific Symposium on Biocomputing*, pp. 219-229, 2017.

[14] K. Sirinukunwattana, S.E.A. Raza, Y.W. Tsang, D.R. Snead, I.A. Cree, and N.M. Rajpoot, "Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images", *IEEE transactions on medical imaging*, 35(5), pp. 1196-1206, 2016.

[15] F. Pala and B. Bhanu, "Iris Liveness Detection by Relative Distance Comparisons", *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 664-671, July 2017.

[16] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", *IEEE International Conference on Artificial Intelligence and Statistics*, pp. 249-256, 2010.

[17] A. Radford, L. Metz and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks", *arXiv:1511.06434*, 2017.

[18] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition", *IEEE computer vision and pattern recognition*, pp. 770-778, 2016.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv:1409.1556*, 2014.

[20] A. Krizhevsky, I. Sutskever and G.E. Hinton, "Imagenet classification with deep convolutional neural networks", *In Advances in neural information processing systems*, pp. 1097-1105, 2012.

[21] B. Bhanu and P. Talbot, "Video Bioinformatics: From Live Imaging to Knowledge", *Springer*, 22, 2015.

[22] R. Mangoubi, C. Jeffreys, A. Copeland, M. Desai and P. Sammak, "Non-invasive image based support vector machine classification of human embryonic stem cells", *IEEE International Symposium on Biomedical Imaging*, pp. 284-287, 2007.