Convolutional Neural Networks for Neonatal Pain Assessment

Ghada Zamzmi, *Member, IEEE*, Rahul Paul, *Member, IEEE*, Md. Sirajus Salekin, *Member, IEEE*, Dmitry Goldgof, *Fellow, IEEE*, Rangachar Kasturi, *Fellow, IEEE*, Thao Ho, and Yu Sun, *Senior Member, IEEE*

***Abstract*—The current standard for assessing neonatal pain is discontinuous and inconsistent because it depends highly on the observers bias. These drawbacks can result in delayed intervention and inconsistent treatment of pain. Convolutional neural networks (CNNs) have gained much popularity in the last decades due to the wide range of its successful applications in medical image analysis, object and emotion recognition. In this paper, we investigated the use of a novel lightweight neonatal convolutional neural network as well as other popular CNN architectures for assessing neonatal pain. We experimented with various image augmentation techniques and evaluated the CNN architectures using two real-world datasets [COPE and neonatal pain assessment dataset (NPAD)] collected from neonates while being hospitalized in the intensive care unit. The experimental results demonstrate the superiority and efficiency of the novel network in assessing neonatal pain. They also suggest that the automatic recognition of neonatal pain using CNN networks is a viable and more efficient alternative to the current assessment standard.**

***Index Terms*—Convolutional neural network, pain assessment, clinical applications, facial expression analysis.**

I. Introduction

For several decades, pediatricians used to believe that neonates do not feel or remember pain [1]. In 1987, the American Academy of Pediatrics (AAP) recognized neonates' sense of pain [1]. Since then many studies (e.g., [2], [3]) reported a strong association between repeated pain exposure (under-treatment) and alterations in the structure and function of the brain. This association has led to the increased use of anesthetic medications such as Morphine and Fentanyl. However, recent studies found that the excessive use (over-treatment) of analgesic medications can cause many side effects [2], [3]. Examples of these effects include a significant increase of feeding intolerance and hypotension. The current standard for assessing neonatal pain is discontinuous and suffers from the intra- and inter-observer variations, which can lead to over- or under-treatment. Therefore, developing intelligent systems that generate continuous and less subjective pain assessment is important to mitigate the shortcomings of the current standard and improve neonatal health care.

A. Overview of Related Work

Various handcrafted based and deep learning based methods were proposed to assess neonatal pain. We provide below an overview of existing neonatal pain assessment methods. A comprehensive review of existing methods and a discussion of current challenges are presented in [4].

1) Pain Assessment Using Handcrafted Features: Nanni *et al.* [5] presented a handcrafted based method to detect facial expressions of pain using different variations of Local Binary Pattern (LBP) descriptor. Specifically, Local Ternary Pattern (LTB), Elongated Local Binary Pattern (ELBP), and Elongated Local Ternary Pattern (ELTP) texture descriptors were applied to COPE infant dataset [6] to extract pain-relevant features. In the pre-processing stage, the images were re-sized, aligned, cropped to obtain the exact facial region, and divided into blocks or cells of 25 × 25. Then, the texture descriptors listed above were applied to these blocks to extract pain-relevant features. To classify the images (204 images) of 26 subjects as pain or no-pain, an ensemble of Radial Basis SVMs was built and evaluated on a testing set. The results showed that ELTP texture descriptor achieved the highest (approx. 0.93) Area Under the Curve of Receiver Operating Characteristic curve (AUC of ROC) as compared to other texture descriptors.

Similarly, Mansor and Rejab [7] presented a LBP-based method for pain assessment that is robust to different level of illuminations. This work modified COPE dataset [6] (26 infants and 204 images) by altering the original images and adding different levels of illuminations. Then, Multi Scale Retinex (MSR) [8] image filter was applied to remove illumination followed by LBP for feature extraction. The extracted texture features were used to train an unsupervised Gaussian classifier and supervised nearest mean classifier. The highest average accuracy (83%) of the proposed method was achieved by the Gaussian classifier.

Celona and Manoni [9] applied a uniform LBP descriptor (P = 8, R = 1, and 59-bins) to static images of COPE dataset after dividing the face region into 25 (5 × 5) non-overlapping regions. To retain the color information, the LBP histogram

Manuscript received September 16, 2018; revised March 18, 2019 and May 13, 2019; accepted May 13, 2019. Date of publication May 27, 2019; date of current version August 6, 2019. This paper was recommended for publication by Associate Editor T. Hassner upon evaluation of the reviewers' comments. *(Ghada Zamzmi and Rahul Paul are co-first authors.) (Corresponding author: Ghada Zamzmi.)*

G. Zamzmi, R. Paul, Md. S. Salekin, D. Goldgof, R. Kasturi, and Y. Sun are with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL 33620 USA (e-mail: ghadh@mail.usf.edu; rahulp@mail.usf.edu; goldgof@mail.usf.edu; r1k@mail.usf.edu; yusun@mail.usf.edu).

T. Ho is with the Morsani College of Medicine Pediatrics Department, University of South Florida, Tampa, FL 33620 USA (e-mail: thaoho@health.usf.edu).

Digital Object Identifier 10.1109/TBIOM.2019.2918619

was computed for each color channel of each region. Then, the texture features extracted for each color of each region were concatenated into a single feature vector that has 4425 dimensions (59 bins × 25 regions × 3 channels). This feature vector was reduced to 175 dimensions using Principal Component Analysis (PCA) followed by L2 normalization. In the final stage, SVM was trained to classify the facial images into pain or no pain. The trained classifier achieved 77.52% average accuracy using the leave-one-subject-out cross validation protocol.

In addition to LBP descriptor, Celona and Manoni [9] applied HOG (Histogram of Oriented Gradients) descriptor to 2 × 2 blocks of 8 × 8 pixel cells with an overlap of half the block and histograms of 9 bins evenly spread from 0 to 180 degrees. Applying this descriptor to 224×224 gray-scale image generates 26244-dimensions feature vector (729 regions × 4 blocks × 9 bins). This feature vector was reduced to 175 dimensions using Principal Component Analysis (PCA) followed by L2 normalization. Using the features extracted by HOG descriptor with SVM achieved 81.75% average accuracy (i.e., accuracies averaged across 26 subjects). Other handcrafted based methods for neonatal pain assessment can be found in [10]–[13].

The challenge of manually designing handcrafted descriptors and extracting the best set of features has motivated researchers to use Convolutional Neural Networks. CNNs learn and extract relevant features, at multiple levels of abstraction, directly from the source data or images. These networks achieved state-of-the-art results in many applications, including clinical and emotion recognition applications.

2) Pain Assessment Using Deep Features: As is well known, training solid CNNs requires large and well-annotated datasets (e.g., ImageNet - approx. 1.2 million images and 1000 classes). In practice, it is restively rare, especially in the medical domain, to find large and well-annotated datasets. Therefore, transfer learning and data augmentation concepts were introduced to handle the lack of data issue.

Transfer learning is the process of applying the knowledge that was learned in one domain to another relevant domain. Recently, transfer learning has become the de-facto method for analyzing medical images because it allows researchers to extract, using models trained on large datasets, relevant features from small medical datasets. Celona and Manoni [9] applied transfer learning method to static images of COPE dataset (26 infants and 204 images) to classify these images as pain or no pain. In particular, the presented method used deep features extracted by a pre-trained CNN (VGG-Face) to train Support Vector Machine (SVM) model. Testing the trained model on unseen data (i.e., leave-one-subject-out cross validation) achieved 82.42% average accuracy. Combining the extracted deep features with the handcrafted features (e.g., LBP) improved the pain classification and yielded an average accuracy of 83.78%.

Zamzmi *et al.* [14] used different pre-trained CNNs to recognize the pain of neonates. Particularly, VGG-Face, which was trained on a face dataset for face recognition, and VGG-F,M,S, which were trained on ImageNet for image classification, were applied to extract deep features from neonates' faces. The features that were extracted by the pre-trained CNNs were used to train Naive Bayes, Nearest Neighbors (kNN), Support Vector Machines (SVMs), and Random Forests (RF). The proposed pipeline achieved up to 90.34% accuracy.

Data augmentation is the process of creating synthetic data, using image processing techniques, that are large enough to train CNN from the scratch. Examples of common image processing techniques that are used for image augmentation include histogram modification, noise addition, zooming in/out, cropping, geometric transformations via random translation, rotation, and flipping, and elastic deformations [15]. In addition to the traditional methods, deep neural networks such as Generative adversarial networks (GANs) are commonly used for image augmentation [16]. Data augmentation is a mandatory pre-processing step for the vast majority of deep learning methods. Presentations of several data augmentation methods can be found in [16], [17].

B. Contributions and Roadmap

CNNs can provide objective pain-relevant features because these networks learn and extract features, at multiple levels of abstraction, directly from the image's pixels. Existing CNNs can perform reasonably well given sufficient training data. However, we believe existing CNNs can not achieve the desired accuracy level for neonatal pain assessment application and similar clinical applications as these networks are trained, using million of images and parameters, for regular image classification.

In this paper, we investigate the use of a novel Neonatal Convolutional Neural Network (N-CNN) for assessing neonatal pain from facial expression [18]. Our N-CNN is designed specifically for analyzing facial expression of neonates. Several studies (e.g., [19], [20], [21], [22]) reported the failure of face recognition methods designed for adults when applied to the neonatal population due to the unique craniofacial structure of neonates' face as well as the large variations in pose and expression as compared to adults. These studies concluded that it is important to design and develop face recognition engines using data collected from neonatal population. Our N-CNN was designed and trained an end-to-end using a real-world dataset of neonates. Our dataset has images that exhibit large variations in pose, appearance, illumination, background and camera viewpoints.

In addition, this paper provides quantitative comparisons between N-CNN and two well-known CNN architectures (VGG and ResNet). We fine-tuned these architectures and evaluated their performance on the same neonatal datasets. We experimented with various image augmentation techniques using both RGB and greyscale images. To the best of our knowledge, we are the first to fully exploit CNNs for assessing pain of neonates recorded in the Neonatal Intensive Care Unit (NICU). The experimental results and the quantitative comparisons demonstrate the superiority and efficiency of the proposed N-CNN in assessing neonatal pain.

Section II presents our Neonatal Pain Assessment Dataset (NPAD). Section III provides a description of N-CNN and brief presentations of two well-known Convolutional Neural Networks: ResNet50 and VGG-16. Section IV presents the experimental procedure, which includes pre-processing, CNNs training, and evaluation protocol. Section V presents the experimental results and quantitative comparisons. Finally, we conclude and discuss different future extensions in Section VI.

II. Neonatal Pain Assessment Dataset (NPAD)

A. Subjects

This study complied with the protocols and ethical directives for research involving human subjects at Tampa General Hospital and the University of South Florida. Prior to the enrollment, a nurse and the principal investigator met the parents to discuss the study objective and explain the procedure. The parents who agreed to allow their baby to participate in the study signed the study's consent form.

Data was collected for a total of 31 Neonates (50% female) in the NICU at Tampa General Hospital. The age of the participating neonates ranged from 32 0/7 to 40 6/7 GW, with a mean age of 35.9 GW. The ethnic distribution was 26% Caucasian, 43% White, 19% African American, and 12% Asian. Any neonate with a gestational age $\geq$ 28 0/7 was eligible for enrollment after obtaining consent from the parents. Neonates with significant facial abnormalities were excluded.

B. Apparatus

We collected video and audio data from neonates while in a baseline state and during painful procedures using a GoPro Hero camera. The camera was triggered remotely using GoPro application installed on a smart device. The recorded data included the neonate's face, head, and body, as well as the sounds of neonates and background noise (e.g., sounds of equipment and nurses). The camera was installed on a stand that faces the neonate's incubator. In addition to the video and audio signals, we collected vital sign readings, namely heart rate (HR), respiratory rate (RR), oxygen saturation levels (SpO2), and blood pressure (BP), from Philips MP-70 monitor. Note that all the data was collected during routine clinical procedures and carried out in the normal clinical environment that was only modified by the addition of the cameras. This makes our dataset highly representative of the real-world condition.

C. Painful Procedure and Ground Truth Labels

All the neonates in our study were recorded during procedural pain. The stimuli that trigger procedural pain are routine heel lancing and immunization. The recording for procedural pain consists of eight time periods: baseline period (T0), procedure preparation period (T1), the painful procedure period (T2), and the post-painful-procedure periods (T3 to T7). The pain assessment for each of these periods was documented by bedside caregivers using the Neonatal Infant Pain Scale (NIPS) [23]. This pain scale consists of facial expression, cry, arms and legs movement, vital signs, and state of arousal. The label for each pain response is 0 or 1 except for cry, which is labeled as 0, 1, or 2. Adding the labels of NIPS components generates a total pain score. The total score is then used to generate, through thresholding, three emotional states or labels:

- No pain state for a score of 0-2.
- Moderate pain state for a score of 3-4.
- Severe pain state for a score greater than 4.

These states, which were documented by trained nurses, are the ground truth labels that were used to train the networks. *The agreement between nurses was measured using Kappa coefficient (0.85) and Pearson correlation (0.89). We included the cases of agreement and excluded the cases of disagreement from further analysis*. It is important to note that because the number of moderate cases in our current dataset is small, we combined the moderate pain cases with severe pain cases and called the combination pain class.

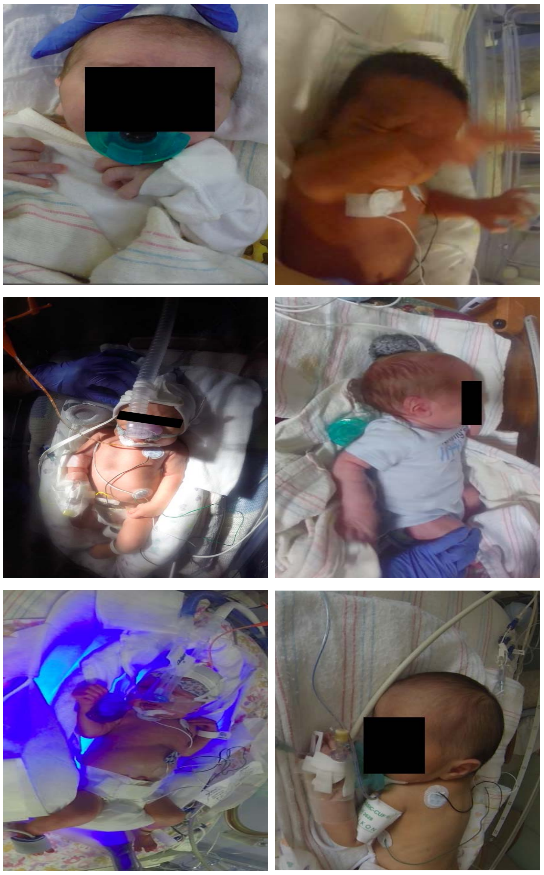

Fig. 1. Image Samples from NPAD dataset.

Figure 1 shows examples of our NPAD dataset. The images were randomly selected and face-masked to ensure confidentiality. As can be seen from the figure, the clinical environment has different level of illumination, pose, and occlusion due to the neonate's hand, pacifier, or oxygen mask. To the best of our knowledge, NPAD is one of the few datasets that is collected specifically for analyzing neonates' behaviors. A portion of NPAD dataset can be accessed, after signing an agreement form, for research use. We refer those who are interested to NPAD dataset's webpage[1] for information about the process of requesting and accessing the dataset.

[1] www.rpal.cse.usf.edu/project_neonatal_pain/dataset.html

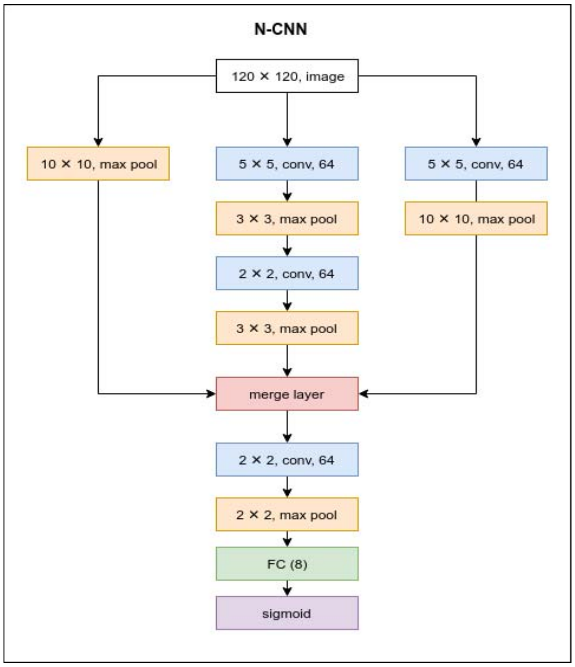

Fig. 2. Overview of N-CNN Architecture [18].

III. Convolutional Neural Networks for Pain Assessment Application

Convolutional Neural Networks (CNNs) have gained much popularity in the last decade due to the wide range of its successful applications in natural language processing, medical image analysis, object recognition, and emotion recognition [24]. The power of CNNs, which are biologically-inspired variants of a multilayer perceptron, can be attributed to its deep architecture that allows to extract a set of features at multiple levels of abstraction directly from the image's pixels. CNN consists of an input layer, an output layer, and three types of hidden layers: convolutional layer, pooling layer, and fully connected layer [24]. This section provides a description of our N-CNN [18] and briefly presents the architectures of ResNet50 and VGG-16, which we utilized in this paper for assessing neonatal pain.

A. Neonatal Convolutional Neural Network (N-CNN)

N-CNN, which was inspired by [25] and proposed in [18], is depicted in Figure 2. As can be seen, N-CNN has a cascaded architecture with three branches, left branch, central branch, and right branch. The left branch applies max pooling filter to the input face image. This branch performs down-sampling operation on the input face image. The central branch has two convolutions layers and two max pooling layers. The convolutions layers of this branch generate feature maps of generic features, such as color blobs and edges, while the pooling layers reduce the spatial size of the generated maps. Finally, the right branch has two layers, a convolutional layer and a pooling layer. This architecture allows to combine the specific information for each image (left branch) with the generic information (edges and blobs) generated after applying convolutional operations. Each branch of N-CNN (Figure 2) takes as input 120 × 120 RGB or greyscale image.

The output of the three branches are then concatenated and sent to a convolution layer followed by a max-pooling layer. As discussed in [18], [25], the cascaded architecture of CNN achieves better classification performance than the regular CNN architecture. The full configuration of N-CNN can be summarized as follows:

- The max-pooling layer of the left branch: 10 × 10 max pooling with 10 stride and 0 padding.
- The first Convolutional layer of the central branch has 64 filters with size 5×5, 1 stride, and 0 padding. This layer is followed by Leaky ReLU (0.01) and a 3×3 max-pooling layer with 3 stride and 0 padding.
- The second Convolutional layer of the central branch has 64 filters with size 2 × 2, 1 stride, and 0 padding. This layer is followed by Leaky ReLU (0.01), a 3 × 3 max pooling layer with 3 stride and 0 padding.
- The first Convolutional layer of the right branch has 64 filters with size 5×5, 1 stride, and 0 padding. This layer is followed by Leaky ReLU (0.01), a 10×10 max-pooling layer (Max-pool 4) with 10 stride and 0.1 dropout.
- The merging layer combines the outputs of the three branches.
- The Convolutional layer after the merging layer has 64 filters with size 2 × 2, 1 stride, and 0 padding.
- The max-pooling layer after the merging layer has 2 × 2 size and 2 stride.
- The first fully connected layer (Fully Connected 1) at the end of the network has 8 units and is followed by Relu, L2 Regularizer (0.01), and Dropout (0.1). L2 Regularizer and Dropout are used to prevent over-fitting.
- The final layer has Sigmoid function that outputs a probability value from 0 to 1.

N-CNN was trained using original and augmented images from NPAD dataset. The total number of epochs for training was 100. We used RMSprop (Root Mean Square Propagation [26]) as a gradient descent optimization algorithm and a constant learning rate of 0.0001. We used a batch size of 16 for both training and validating N-CNN; note that we have experimented with different batch sizes (8/16/24/32/40) and chose 16 since it achieved the best performance [18]. To prevent over-fitting, L2 regularizer [27] and dropout [28] were applied before the final classification layer [18].

B. Visual Geometry Group Convolutional Neural Network (VGG-16)

VGG-16 [29], which was trained using ImageNet dataset, has 16 layers. It passes an input RGB image with size 224 × 224 to a stack of 13 convolutional layers, each uses a small filter of size 3 × 3 with 1 stride and 1 padding to extract features. Five max pooling with 2 × 2 window and stride 2 are used after each block of the convolutional layers. The stack of convolutional and max pooling layers is followed by three fully connected layers and a softmax layer. The fully connected layers have 4096, 4096, and 1000 units, respectively. The number of units in the last layer corresponds to the number of classes in ImageNet dataset (1000 classes). All the hidden layers are equipped with ReLU function. VGG-16 is trained using 138 millions parameters. The complete list of training and implementation parameters can be found in [29].

C. Deep Residual Convolutional Neural Network (ResNet 50)

ResNet 50 [30], which was trained using ImageNet dataset, has 50 layers. The network takes as input 224 × 224 RGB image and passes the given image to a 7 × 7, 64 convolutional layer with stride 2 followed by 3×3 max pooling with stride 2. The output is then sent to a stack of 48 convolutional layers distributed over four blocks. Each block starts with a convolutional layer that has a filter size of 1 × 1 followed by a convolutional layer that has a filter size of 3 × 3 and ends with a convolutional layer that has 1 × 1 filter size. The stack of the convolutional layers (the blocks) is followed by a fully connected layer with 1000 units (1000 classes) and a softmax layer. ResNet50 was trained using augmented images (scale and color augmentation [30]) from ImageNet dataset with 25.6 million parameters. The complete list of training and implementation parameters of ResNet50 can be found in [30].

IV. Experimental Procedure

The experimental procedure can be divided into: pre-processing, CNNs training, and model evaluation. Each of these stages is presented next.

A. Pre-Processing

The pre-processing stage involves detecting the face region and performing augmentation to increase the size of the dataset.

To create the training and testing sets from the given video sequences, we extracted the key frames, thereby removing many similar frames, from each video sequence after cropping the face using ZFace tracker [31]. The total number of key frames obtained from all the videos of 31 subjects was 3026 frames. These frames were divided into a training set and a testing set.

To increase the size of the training set, we performed two types of image augmentation: geometric augmentation and elastic augmentation. In the geometric augmentation, we rotated each image by 30 degrees followed by flipping the rotated image horizontally and vertically. This process generated a total of 36 augmented images for each image (12 rotated + 12 vertical-flip + 12 horizontal-flip). Different studies reported classification improvements when models trained using elastically deformed images [15], [32], [33]. Therefore, we applied, using the toolbox of [34], elastic deformations to the training set as follows [15], [34]:

$$F(x_1, y_1) = m \times I(x, y) + D(a, b)$$

where F is the new location of the original pixel, m is the strength of the displacement, I is the original pixel value, and D is the displacement vector. The values of variables a, b, and m were chosen empirically such that the similarity (Structural Similarity Index) between original and augmented images is kept less than 85%. Both a and b were fixed to be 5 and three different values for m (3, 4, 5) were chosen. This process generates 3 elastically augmented images for each image in the training set. Figure 3 shows examples of geometrically and elastically augmented images.

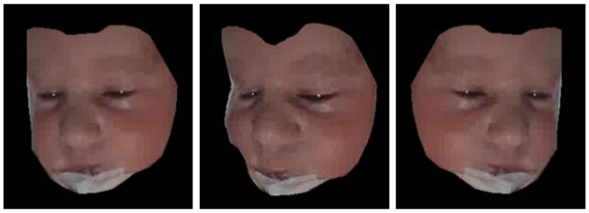

Fig. 3. Examples of Augmented Images. First column: original. Second column: 5,5,4 elastic deformation. Third column: flipped original.

B. Feature Representation Using CNN Architectures

We trained N-CNN, VGG-16, and ResNet50 using facial images from NPAD dataset, which was collected under large variations in pose and illumination. All the networks were trained using original training set, geometrically augmented training set, and elastically augmented training set. The training parameters for all the networks are presented below.

1) Training N-CNN Architecture: We trained N-CNN with random weights initialization and 72593 training parameters. The total number of epochs for training N-CNN was 100. We used RMSprop (Root Mean Square Propagation [26]) as the gradient descent optimization algorithm and a constant learning rate of 0.0001. We used a batch size of 16 for both training and validating N-CNN. We applied L2 regularizer [27] and dropout [28] before the final classification layer to prevent over-fitting. Note that we experimented with different image sizes, specifically 214×214, 120×120, 100×100, and chose 120×120 because it achieved the best performance. Similarly, we experimented with different batch sizes (8/16/24/32/40) and chose 16 since it achieved the best performance.

2) Training VGG-16 and ResNet50 Architectures: We fine-tuned VGG-16 and ResNet50 using images from NPAD dataset. For both architectures, we used the ImageNet trained weights in the lower layers and only changed the upper layers' parameters using NPAD dataset. We fine-tuned VGG-16 and ResNet50 as shown in Table I and Table II. The total number of parameters for the tuned VGG16 and ResNet50 architectures are 27,823,425 and 23,688,065 parameters, respectively. As can be seen in Table III, we changed the layer parameters of VGG-16 and added dropout of 0.5 after each of the fully connected layers to reduce over-fitting. In case of ResNet50, we used global average pooling to obtain the base model output from the lower layers of ResNet-50. We also added a dropout of 0.5 after the global average pooling to reduce overfitting. We used RMSprop (Root Mean Square Propagation [26]) as the gradient descent optimization algorithm, a constant learning rate of 0.0001, and batch size of 16. Note that the number of training parameters for N-CNN (72593) is significantly lower than the number of training parameters for the tuned ResNet50 (27,823,425) and VGG-16 (23,688,065).

TABLE I
TUNED VGG-16 ARCHITECTURE

Conv3	64 × 3 × 3, st. 1, pad 1
Conv 1-2	64 × 3 × 3, st. 1, pad 1
Conv 2-1	128 × 3 × 3, st. 1, pad 1
Conv 2-2	128 × 3 × 3, st. 1, pad 1
Conv 3-1	256 × 3 × 3, st. 1, pad 1
Conv 3-2	256 × 3 × 3, st. 1, pad 1
Conv 3-3	256 × 3 × 3, st. 1, pad 1
Conv 4-1	512 × 3 × 3, st. 1, pad 1
Conv 4-2	512 × 3 × 3, st. 1, pad 1
Conv 4-3	512 × 3 × 3, st. 1, pad 1
Conv 5-1	512 × 3 × 3, st. 1, pad 1
Conv 5-2	512 × 3 × 3, st. 1, pad 1
Conv 5-3	512 × 3 × 3, st. 1, pad 1
Full 6	512, dropout =0.5 , relu
Full 7	512, dropout =0.5, relu
Full 8	1, sigmoid
Total Parameters	27,823,425

TABLE II
TUNED RESNET50 ARCHITECTURE

Global Average Pooling	Base model output
Dropout	0.5
Full 1	1, sigmoid
Total Parameters	23,688,065

C. Evaluation Protocol

After training N-CNN, VGG-16, and ResNet50 using NPAD images, we evaluated the trained networks on unseen data and compared the performance of these networks against each other. In our previous work [18], we randomly split the dataset into a training set and a testing set three times, built the network using the training set, and evaluated the trained network on the testing set. We then reported the averaged performance of all iterations as the final performance. In this paper, we used leave-one-subject-out cross validation evaluation protocol. This protocol is more realistic because the neonate's hospitalization period is usually short. That means, in real life, the trained model will be tested, out of the box, on an unseen set of future subjects. We performed leave-one-subject-out cross validation as follows. We divided the entire dataset into 31 folds corresponding to 31 subjects. In each splitting, we used the images of one subject for testing and the images of remaining subjects for training. We repeated this process 31 times and the accuracies of the 31 folds were averaged to obtain the final performance. In each iteration, all the images that belong to one subject are either in the training set or testing set, but not in both.

V. EXPERIMENTAL RESULTS

This section presents the results of assessing neonatal pain using N-CNN, VGG-16, and ResNet50 architectures. We trained these networks using either the original set, geometrically augmented set, or elastically augmented set. We then used leave-out-subject-out cross validation protocol to evaluate the performance of the trained networks. This section also reports the performance of N-CNN using a grey-scale version of NPAD dataset as well as the performance of evaluating N-CNN on another publicly available neonatal dataset. We report the performance using the values of the confusion matrix, namely the accuracy, true positive rate (TPR), true negative rate (TNR), false positive rate (FPR), and false negative rate (FNR).

A. Pain Assessment: Original Images

The first column of results in Table III reports the performance of assessing pain using N-CNN, ResNet50, and VGG-16 with the original dataset (no augmentation). We followed this direction because we wanted to evaluate how the augmentation affects the classification performance along with the true positive rate (TPR) and false positive rate (FPR). As shown in Table III, the highest accuracy presented in the first column (no augmentation) is obtained by the proposed N-CNN. The performance difference between N-CNN (93.47%) and ResNet50 (85.83%) is statistically significant (P value = 0.05). However, the performance difference between N-CNN and VG-16 is not statistically significant (P value = 0.05). By comparing the first column of Table III with the second and third columns, we can see that using the original training set without augmentation yielded the lowest performance for N-CNN, ResNet50, and VGG-16.

B. Pain Assessment: Augmented RGB Images

The second column of results in Table III shows the assessment performance of the three CNNs when trained using a geometrically augmented training set. By comparing the performance of the first and second columns in Table III, we can see that training N-CNN and VGG-16 using geometrically augmented set significantly improves the overall performance as well as the performance of the pain class. The pain assessment accuracy, TNR, and FPR improve when ResNet50 trained using geometrically augmented set. However, the performance of the pain class (TPR and FNR) increased slightly. Therefore, we can conclude that N-CNN achieved the best overall performance as it has the highest accuracy and lowest FPR and FNR. It is worth mentioning that minimizing both the FPR and FNR rates is equally important in case of pain assessment as pediatric studies reported serious outcomes of both over-treatment (FPR) and under-treatment (FNR). The last column of results in Table III provides the performance of assessing pain using an elastically augmented training set. As can be seen, the performance of assessment using the elastically augmented set is higher than the no-augmentation set for all CNNs. For all the three CNNs, training the networks using the geometrically augmented set achieve better performance than training using the elastically augmented set.

Table IV presents the performance of the three CNNs with the original set, the geometrically augmented set, and the elastically augmented set for all subjects in NPAD dataset. The presented results show that the pain assessment performance achieved by the proposed N-CNN is comparable to, if not better than, ResNet50 and VGG-16. Note that N-CNN has the lowest number of parameters as compared to VGG16 and ResNet50. These results are encouraging and demonstrate the superiority and efficiency of the proposed N-CNN in pain assessment application, and possibly, similar applications.

TABLE III
PERFORMANCE OF PAIN ASSESSMENT USING N-CNN, RESNET50, AND VGG-16

	Original (No Augmentation)					Geometric Augmentation					Elastic Deformation				
	Acc	TPR	TNR	FPR	FNR	Acc	TPR	TNR	FPR	FNR	Acc	TPR	TNR	FPR	FNR
N-CNN	**93.47**	83.19	96.55	3.45	16.81	**96.98**	91.5	98.00	**2.0**	**8.5**	**96.39**	87.00	98.00	2.00	13.00
ResNet50	85.83	81.26	87.27	12.73	18.74	93.77	82.64	97.25	2.75	17.69	91.8	81.95	95.03	4.97	18.05
VGG-16	93.27	83.61	96.16	3.84	16.39	95.69	89.53	97.64	2.36	10.47	91.1	87.46	96.21	3.79	12.54

TABLE IV
PERFORMANCE PER SUBJECT AND AVERAGE PERFORMANCE FOR N-CNN, VGG-16, AND RESNET50

	No Augmentation			Elastic Deformation			Geometric Augmentation		
	N-CNN	VGG-16	ResNet50	N-CNN	VGG-16	ResNet50	N-CNN	VGG-16	ResNet50
Subject 1	88%	84%	93%	88%	86%	93%	95%	88%	95%
Subject 2	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 3	39%	47%	42%	68%	66%	60%	66%	66%	66%
Subject 4	93%	74%	75%	93%	74%	76%	91%	76%	76%
Subject 5	85%	52%	81%	94%	60%	82%	100%	81%	85%
Subject 6	76%	100%	73%	100%	100%	76%	100%	100%	100%
Subject 7	97%	90%	95%	100%	90%	95%	100%	95%	97%
Subject 8	76%	93%	79%	85%	93%	80%	90%	93%	80%
Subject 9	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 10	100%	81%	100%	100%	86%	100%	100%	100%	100%
Subject 11	69%	88%	90%	96%	90%	96%	96%	96%	96%
Subject 12	98%	100%	86%	98%	100%	89%	89%	100%	89%
Subject 13	81%	81%	96%	90%	81%	96%	100%	90%	96%
Subject 14	93%	72%	86%	94%	88%	88%	97%	93%	93%
Subject 15	100%	100%	97%	100%	100%	98%	100%	100%	100%
Subject 16	100%	100%	97%	100%	100%	100%	100%	100%	100%
Subject 17	52%	74%	52%	62%	74%	62%	67%	74%	67%
Subject 18	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 19	96%	98%	91%	98%	98%	96%	98%	98%	98%
Subject 20	100%	95%	68%	98%	91%	68%	100%	95%	75%
Subject 21	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 22	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 23	93%	91%	95%	98%	91%	95%	95%	95%	95%
Subject 24	93%	93%	93%	93%	94%	93%	93%	94%	94%
Subject 25	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 26	99%	100%	94%	100%	100%	100%	100%	100%	100%
Subject 27	99%	100%	98%	100%	100%	100%	100%	100%	100%
Subject 28	100%	100%	100%	100%	100%	100%	100%	100%	100%
Subject 29	91%	92%	95%	92%	96%	96%	94%	96%	96%
Subject 30	100%	93%	100%	100%	95%	100%	100%	97%	100%
Subject 31	97%	100%	91%	100%	100%	94%	100%	100%	97%
Average	**93.5%**	93.3%	85.8%	**96.4%**	94.1%	91.9%	**97.0%**	95.7%	93.8%

TABLE V
PERFORMANCE OF NCNN ON GRAYSCALE AND RGB IMAGES

	Grayscale Images						RGB Images					
	Acc.	AUC	TPR	TNR	FPR	FNR	Acc.	AUC	TPR	TNR	FPR	FNR
No Augmentation	84.0%	0.72	34.6%	99.6%	0.39%	65.4%	93.5%	0.88	83.2%	96.6%	3.5%	16.8%
Geometric Augmentation	**94.4%**	0.88	77.7%	99.7%	0.3%	22.3%	**97.0%**	0.94	91.5%	98%	2%	8.5%
Elastic Deformation	91.0%	0.82	63.5%	99.7%	0.3%	36.5%	96.4%	0.93	87%	98%	2%	13%

C. Pain Assessment: Augmented Greyscale Images

To understand the impact of color information on pain assessment performance, we trained N-CNN using a greyscale version of NPAD dataset. We first converted the RGB images of NPAD dataset into greyscale images. Then, we trained N-CNN three times using: 1) original training set, 2) geometrically augmented training set, and 3) elastically augmented training set. We used leave-one-subject-out cross validation for evaluation. Table V presents the pain assessment performance using RGB images and greyscale images from NPAD dataset. We reported the performance using measures of confusion matrix as well as the Area Under the Curve of Receiver Operating Characteristic curve (AUC of ROC). As the table shows, using color information significantly improves the pain assessment performance for all sets. In particular, the difference of AUC between greyscale and RGB images for all three sets is statistically significant ($P = 0.05$). These results show that color information is important for pain assessment application, and possibly for similar applications.

D. Evaluation on COPE Dataset

To further evaluate N-CNN, we tested the trained network on another publicly available dataset known as COPE dataset [6], [18]. To the best of our knowledge, COPE (and

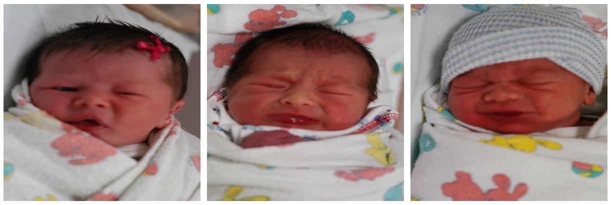

Fig. 4. First column: image labeled as pain in COPE, but classified as no-pain. Second and third columns: images labeled as no-pain in COPE, but classified as pain.

iCOPE) is the only dataset of neonates that is currently available for research use. We applied the trained CNNs (N-CNN, ResNet50, and VGG-16) on static images of COPE dataset and reported the results. COPE dataset [6] consists of 204 static images taken during four different stimuli: 1) pain stimulus, 2) rest/cry stimulus, 3) air stimulus to the nose, and 4) friction stimulus. We divided all the images of COPE dataset into: pain set (heel-lancing) and no-pain set (other stimuli). Applying the trained (geometrically augmented set) N-CNN, ResNet50, and VGG-16 on COPE achieved 89.8%, 85.4%, and 88.2% accuracies, respectively. As expected, N-CNN achieved the highest performance as compared to ResNet50 and VGG-16. Figure 4 shows examples of the cases that were misclassified by the three CNNs.

E. Pain Assessment: Handcrafted Methods

To establish a firm baseline and compare the performance of well-known handcrafted methods with N-CNN, we applied two handcrafted methods to NPAD dataset: Local Binary Pattern (LBP) and Histogram of Oriented Gradients (HOG). These methods were used to extract features from neonates' faces followed by classification using machine learning classifiers. Both methods were evaluated using leave-one-subject-out cross validation.

In case of LBP, we applied LBP appearance descriptor to small patches of the neonate's face to extract pain-relevant features. We extracted, using LBP, appearance features from 32×32 patches located around 31 facial landmark points. The feature vector was reduced and then used to train SVM classifier. The output of the classifier is a binary label that indicates if the pain is present or not. Assessing neonatal pain using LBP features achieved 86.8% average accuracy. Similarly, we extracted Histogram of oriented gradients (HOG) features. The extracted feature vector was reduced and used to train different machine classifiers. The output of the classifier is a binary label that indicates if the pain is present or not. Assessing neonatal pain using HOG features with SVM achieved 81.29% average accuracy. We refer the reader to [11], [13], [35] for presentation of pain assessment performance using other handcrafted methods with NPAD dataset.

To summarize, we compare above the performance of N-CNN with two handcrafted methods (LBP and HOG) and two well-known CNNs (ResNet50 and VGG-16). We reported the performance on two neonatal datasets: NPAD and COPE. Our proposed N-CNN, which extracts features directly from the images, achieved state-of-the-art results and outperformed ResNet, VGG-16 as well as handcrafted descriptors. The proposed N-CNN has the lowest number of parameters as compared to both VGG16 and ResNet50 (N-CNN: 72593, VGG16: 27,823,425, and ResNet50: 23,688,065). These results are encouraging and demonstrate the superiority and efficiency of using the proposed N-CNN for pain assessment application, and possibly, similar applications.

VI. Conclusion and Ongoing Works

The current practice for assessing neonatal pain is inconsistent because it depends highly on the observer's bias. Additionally, it is discontinuous and requires a large number of well-trained nurses to ensure the proper utilization of the pain scale. The discontinuous nature of the current practice as well as the intra- and inter-observer variations may result in delayed intervention and inconsistent treatment of pain. Since pain assessment is the cornerstone of pain management, developing automatic and continuous scales that generate immediate and more consistent pain assessment is crucial.

This paper investigates the use of a novel Neonatal CNN (N-CNN) along with other two well-known CNNs (ResNet and VGG-16) for pain assessment application. All the networks were evaluated using a real-world dataset collected from 31 neonates hospitalized in the NICU. To the best of our knowledge, this paper is the first to fully exploit the use of different CNN architectures for pain assessment application. The experimental results showed that the pain assessment performance achieved by N-CNN is comparable to, if not better than, ResNet50 and VGG-16. These results are encouraging and suggest that the automatic recognition of neonatal pain is a viable and more efficient alternative to the current standard of pain assessment. By continuing to explore the use of CNNs for developing a highly accurate pain assessment application, we hope to improve the effectiveness of pain intervention while mitigating the short- and long-term outcomes of pain exposure in early life.

Ongoing work includes integrating other pain indicators such as body movement and crying sound to facial expression to obtain a multimodal network for neonatal pain assessment. The multimodal approach for pain assessment is necessary because it allows to assess pain during circumstances when not all pain responses are available. Examples of these circumstances include occlusion (e.g., prone position or full oxygen mask), clinical condition (e.g., Bell's palsy), level of activity (e.g., physical exertion), and sedation. In addition, we are working on integrating contextual information, such as medication type/dose, to obtain a context-sensitive pain assessment. Because integrating this information requires a large dataset, we are currently involved in an ongoing effort to collect a large multimodal dataset from a couple of hundred neonates during their hospitalization in the NICU. Finally, we plan to investigate the possibility of using neonates' sounds as soft biometric since our preliminary results suggest that each neonate might have a unique sound pattern.

Acknowledgment

The authors are grateful to the parents who allowed their babies to take part in this study and the entire neonatal staff at Tampa General Hospital for their help and cooperation in the data collection. The authors are also grateful for the helpful feedback from the anonymous reviewers of this paper.

Ghada Zamzmi and Rahul Paul contributed equally to the design and implementation as well as the experiments and the analysis of the results. Md Sirajus Salekin performed a portion of the experiments (grayscale, original) and helped in preparing the results. Dmitry Goldgof, Rangachar Kasturi, Thao Ho, and Yu Sun supervised the research. Ghada Zamzmi wrote the paper with input from all authors.

References

[1] A. Marchant, "'Neonates do not feel pain': A critical review of the evidence," *Biosci. Horizons Int. J. Student Res.*, vol. 7, pp. 1–9, Sep. 2014.

[2] M. D. Cruz, A. M. Fernandes, and C. R. Oliveira, "Epidemiology of painful procedures performed in neonates: A systematic review of observational studies," *Eur. J. Pain*, vol. 20, no. 4, pp. 489–498, 2016.

[3] T. Field, "Preterm newborn pain research review," *Infant Behav. Develop.*, vol. 49, pp. 141–150, Nov. 2017.

[4] G. Zamzmi, R. Kasturi, D. Goldgof, R. Zhi, T. Ashmeade, and Y. Sun, "A review of automated pain assessment in infants: Features, classification tasks, and databases," *IEEE Rev. Biomed. Eng.*, vol. 11, pp. 77–96, 2017.

[5] L. Nanni, S. Brahnam, and A. Lumini, "A local approach based on a local binary patterns variant texture descriptor for classifying pain states," *Expert Syst. Appl.*, vol. 37, no. 12, pp. 7888–7894, 2010.

[6] S. Brahnam, C.-F. Chuang, F. Y. Shih, and M. R. Slack, "Machine recognition and representation of neonatal facial displays of acute pain," *Artif. Intell. Med.*, vol. 36, no. 3, pp. 211–222, 2006.

[7] M. N. Mansor and M. N. Rejab, "A computational model of the infant pain impressions with Gaussian and nearest mean classifier," in *Proc. IEEE Int. Conf. Control Syst. Comput. Eng.*, 2013, pp. 249–253.

[8] Z.-U. Rahman, D. J. Jobson, and G. A. Woodell, "Multi-scale retinex for color image enhancement," in *Proc. 3rd IEEE Int. Conf. Image Process.*, vol. 3, 1996, pp. 1003–1006.

[9] L. Celona and L. Manoni, "Neonatal facial pain assessment combining hand-crafted and deep features," in *Proc. Int. Conf. Image Anal. Process.*, 2017, pp. 197–204.

[10] G. Zamzami, G. Ruiz, D. Goldgof, R. Kasturi, Y. Sun, and T. Ashmeade, "Pain assessment in infants: Towards spotting pain expression based on infants' facial strain," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, vol. 5, 2015, pp. 1–5.

[11] G. Zamzmi, C.-Y. Pai, D. Goldgof, R. Kasturi, Y. Sun, and T. Ashmeade, "Automated pain assessment in neonates," in *Proc. Scandinavian Conf. Image Anal.*, 2017, pp. 350–361.

[12] E. Fotiadou, S. Zinger, W. E. T. A. Ten, S. B. Oetomo, and P. H. N. de With, "Video-based facial discomfort analysis for infants," in *Proc. SPIE*, 2014, p. 9029. doi: 10.1117/12.2037661.

[13] R. Zhi, G. Zamzmi, D. Goldgof, T. Ashmeade, and Y. Sun, "Automatic infants' pain assessment by dynamic facial representation: Effects of profile view, gestational age, gender, and race," *J. Clin. Med.*, vol. 7, no. 7, p. 173, 2018.

[14] G. Zamzmi, D. Goldgof, R. Kasturi, and Y. Sun, "Neonatal pain expression recognition using transfer learning," *arXiv preprint arXiv:1807.01631*, Jul. 2018.

[15] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?" in *Proc. Int. Conf. Digit. Image Comput. Techn. Appl. (DICTA)*, 2016, pp. 1–6.

[16] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," arXiv preprint arXiv:1712.04621, Dec. 2017.

[17] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.

[18] G. Zamzmi, R. Paul, D. Goldgof, K. Rangachar, and Y. Sun, "Pain assessment from facial expression: Neonatal convolutional neural network (N-CNN)," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–6.

[19] S. Bharadwaj, H. S. Bhatt, M. Vatsa, and R. Singh, "Domain specific learning for newborn face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 7, pp. 1630–1641, Jul. 2016.

[20] D. Wen, C. Fang, X. Ding, and T. Zhang, "Development of recognition engine for baby faces," in *Proc. 20th Int. Conf. Pattern Recognit.*, Istanbul, Turkey, 2010, pp. 3408–3411.

[21] Z. Hammal, W.-S. Chu, J. F. Cohn, C. Heike, and M. L. Speltz, "Automatic action unit detection in infants using convolutional neural network," in *Proc. 7th Int. Conf. Affective Comput. Intell. Interact. (ACII)*, 2017, pp. 216–221.

[22] R. E. Grunau, T. Oberlander, L. Holsti, and M. F. Whitfield, "Bedside application of the neonatal facial coding system in pain assessment of premature infants," *Pain*, vol. 76, no. 3, pp. 277–286, 1998.

[23] D. Hudson-Barr, B. Capper-Michel, S. Lambert, T. Mizell Palermo, K. Morbeto, and S. Lombardo, "Validation of the pain assessment in neonates (pain) scale with the neonatal infant pain scale (NIPS)," *Neonatal Netw.*, vol. 21, no. 6, pp. 15–21, 2002.

[24] S. Haykin, *Neural Networks*, vol. 2. New York, NY, USA: Prentice-Hall, 1994.

[25] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, 2015, pp. 5325–5334.

[26] T. Tieleman and G. Hinton, "Lecture 6.5-RMSProp: Divide the gradient by a running average of its recent magnitude," *COURSERA Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.

[27] A. Y. Ng, "Feature selection, L1 vs. L2 regularization, and rotational invariance," in *Proc. ACM 21st Int. Conf. Mach. Learn.*, 2004, p. 78.

[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

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

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[31] L. A. Jeni, J. F. Cohn, and T. Kanade, "Dense 3D face alignment from 2D videos in real-time," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, vol. 1, 2015, pp. 1–8.

[32] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Proc. IEEE 7th Int. Conf. Doc. Anal. Recognit.*, 2003, p. 958.

[33] L. Roose, W. De Maerteleire, W. Mollemans, and P. Suetens, "Validation of different soft tissue simulation methods for breast augmentation," *Int. Congr. Series*, vol. 1281, pp. 485–490, May 2005.

[34] M. D. Bloice, C. Stocker, and A. Holzinger, "Augmentor: An image augmentation library for machine learning," arXiv preprint arXiv:1708.04680, Aug. 2017.

[35] G. Zamzmi, P. Chih-Yun, D. Goldgof, R. Kasturi, T. Ashmeade, and Y. Sun, "A comprehensive and context-sensitive neonatal pain assessment using computer vision," *IEEE Trans. Affect. Comput.*, to be published.