Subject Skin Tone Classification with Implications in Wound Imaging using Deep Learning

Masrur Sobhan[#]
Computing and Information Sciences
Florida International University
Miami, USA
msobh002@fiu.edu

Daniela Leizaola[#]
Biomedical Engineering
Florida International University
Miami, USA
dleiz001@fiu.edu

Anuradha Godavarty [*]
Biomedical Engineering
Florida International University
Miami, USA
godavart@fiu.edu

Ananda Mohan Mondal[*]
Computing and Information Sciences
Florida International University
Miami, USA
amondal@fiu.edu

***Abstract* — Chronic wound healing is inconsistent on an individual basis, leading to large treatment costs. The effectiveness of any treatment approach is typically assessed by visual inspection of the wound. Optical imaging technologies have recently been developed to objectively assess wound physiology to complement the subjective visual assessment. One such device is a low-cost SmartPhone Oxygenation Tool (SPOT), which can measure the tissue oxygenation of the wounds via non-contact imaging and assessing healing status. The varying skin tones impact tissue oxygenation measurements due to the different melanin concentrations in the epidermis of the skin. Hence, it is essential to consider melanin-related attenuation in the epidermis and account for it during tissue oxygenation measurements. This study aims to implement a machine learning algorithm to classify the skin tones using in-vivo measurements from control subjects towards a future correction for these skin tones during imaging studies using SPOT.**

In this study, we developed a benchmark dataset of 75,348 samples of 28 × 28 RGB images of human subjects' hands. The images were then converted to 28 × 28 grayscale images and were flattened to attain 784 pixels or features for each sample. We also developed a deep learning-based pipeline to classify the FST skin types, producing high accuracies (> 98%). The deep learning model can be incorporated into the SPOT device as an additional feature to verify or correct the melanin concentration during near-infrared (NIR) imaging of wound regions.

Keywords— Deep learning, chronic wounds, Fitzpatrick skin tone, hyperparameter tuning, melanin concentration

Type of Submission: "Full/Regular Research Paper"

[*]Corresponding author; [#]Equal contribution.

I. INTRODUCTION (*HEADING 1*)

In 2014, 8.2 million Medicare beneficiaries were affected by chronic non-healing lower extremity wounds. Infection management resulted in an expenditure of $28.1 billion as principal diagnosis up to $96.8 billion with secondary diagnosis inclusive [1]. Of the twelve wound types considered in the cost analysis, four ulcer types were included: venous stasis ulcers, arterial/ischemic ulcers, diabetic/neurotrophic ulcers, and pressure ulcers. Despite the ample availability of lower extremity ulcer therapies, the effectiveness of chronic wound healing is not consistent on an individual basis; hence, the large treatment cost [2], [3]. Clinicians typically evaluate multiple therapies until they determine the most effective for the patient/wound. The treatment's response is assessed by visual inspection of the wound [4]. Typical analysis includes the wound bed size, shape, and depth, as well as a vascular evaluation. For severely infected ulcers, an x-ray and magnetic resonance imaging (MRI) may be necessary for a more thorough review [4]. With the subjectivity of the clinician's visual inspection and the high-cost imaging modalities, low-cost optical technologies have been developed to promote objective interventions for ulcer progression.

An optical modality recently applied for assisting in diagnosing wound healing is near-infrared spectroscopy (NIRS). Due to the non-invasiveness of the modality, infected ulcers can be evaluated in a non-contact and non-ionizing manner. NIR light is minimally absorbed and preferentially scattered; thus, the technique allows imaging of major components in deep tissue (i.e., oxy- and deoxy-hemoglobin). A smartphone-based dual-wavelength NIR device, SmartPhone Oxygenation Tool (SPOT), has been developed to determine oxygenation measurements of physiological tissue. Multiple in-vivo studies have been conducted for area-based imaging of control subjects [5]–[8] and diabetic foot ulcer wounds [5] to exhibit the potential of the technology for wound healing assessments. Nonetheless, variations in the skin pigmentation (i.e., melanin concentrations) surrounding the ulcer bed can affect the oxygenation map of the wound area, previously not accounted for.

Skin tone can be classified by the Fitzpatrick skin typing (FST) system from I-VI (light to dark), where a positive correlation is noted between FST and melanin (chromophore responsible for pigmentation) within the epidermis. NIR optical imaging applied to dermatological diseases involve a range of pigmentation within the area; therefore, it is essential to consider the contribution of this melanin-related attenuation (in the epidermis). With the consideration of skin pigmentation on area-based oxygenation maps, the SPOT device can be used among diverse racial/ethnic groups.

Hence, the objectives of the current pilot study are (i) to obtain digital color images of in-vivo tissues with varying FST scales (I-VI) and (ii) to implement machine learning algorithms to detect the skin tone automatically. In our prior work [9], [10], we explored six algorithms, including Least Absolute Shrinkage and Selection Operation (LASSO) [11], Multi-Cluster Feature Selection (MCFS) [12], Random Forest (RF) [13], Support Vector Machine with Recursive Feature

979-8-3503-2028-2/22/$31.00 ©2022 IEEE
DOI 10.1109/CSCI58124.2022.00331

Elimination (SVMRFE) [14], Unsupervised Discriminative Feature Selection (UDFS) [15], and a deep learning algorithm Multi-run Concrete Autoencoder (mrCAE) [16] on a dataset of 9,000 FST skin tone samples (obtained from colored images of a standard FST scale sticker) with 300 features. Among the 300 features, we identified six sets of the best 20 features, which produced a near-perfect (~100% accuracy) classification of six skin tone types.

By the motivation of our previous studies on FST scale stickers, we first developed a benchmark dataset for skin tone types from pilot in-vivo data from control subjects. Then we developed a deep learning framework to classify different skin tone types. In our future work, the developed ML framework will be tested on an extensive database of control subjects to determine the efficacy of the ML algorithms before using it to determine skin tone and correct for the corresponding melanin concentration during NIR imaging of wounds using the SPOT device.

II. Methodology

A. Dataset Generation and Processing

Fig. 1 outlined the process of dataset generation and processing for Fitzpatrick skin tone types (I-VI).

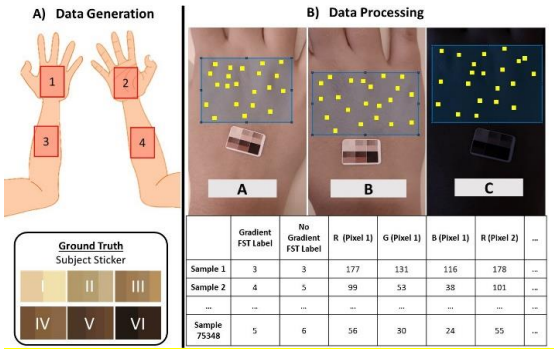

Fig. 1. **Dataset generation and processing**. (a) Data generation: Color images of 65 frames of subjects were taken at three light levels: Level A: White Light with Yellow Lamp, Level B: White Light, Level C: Ambient. Twenty (20) ROIs are randomly generated as 28×28 pixel regions for each skin type (I-VI), across each frame (1-65) and for each light level. (b) Data Processing: Each ROI of 28×28 pixels represents a sample with 2,352 (28×28×3) RBG features labeled with the corresponding skin tone type (I-VI).

Data generation: The color images of 65 frames of 4 subject locations (dorsal/ventral hand and dorsal/ventral arm) were taken at three light levels: Level A: White light with a yellow lamp, Level B: White light, and Level C: Ambient light. A total of 5 subjects were imaged and the images were labeled with a Fitzpatrick Skin Types (FST) as the ground truth. Previous work involved categorization of 6 FST based on 6 color tones [9], [10]. Since there are skin tone variations among subjects, this study utilized a ground truth with multiple tones within each of the 6 FSTs. With the samples from all locations included, the study involved skin types I-VI based on FST scale, shown in Fig. 1A. Twenty regions of interest (ROIs) were randomly generated as 28×28-pixel regions across each frame (1-65), and for each light level, Fig 1B.

Data processing: Each ROI is a sample labeled with the corresponding skin type based on the FST ground truth. The samples were uniformly distributed across the subjects, each having 15,600 (20 ROIs/frame × 65 frames/light condition × 3 light conditions/location × 4 locations) samples, thus ideally generating a total of 78,000 samples. Due to an imaging error in the case of locations 1 and 4, a total of 75,348 samples were finally processed. Each sample with 28×28 pixels of 3 channels (R, G, B) was converted into grayscale using the following expression [17].

$$(0.299 \times R + 0.587 \times G + 0.114 \times B) \qquad \text{Eq. 1}$$

Thus, the final dataset used for the feature selection algorithm consists of a matrix with 75,348 samples × 784 features.

B. Data Distribution across skin tones

The dataset is not balanced across all skin tones due to the variability of skin tone pigmentation or melanin levels of the subjects. Table I shows the overall distribution of data.

TABLE I. **Distribution of skin tone data.** The table shows the data distribution in four different locations based on the gradient FST scaling. FST III has the highest number of samples and FST VI has the lowest number of samples.

Location	Gradient FST Scaling	Sample Size
Dorsal Hand	III	10,400
	IV	3.900
	V	3.900
Ventral Hand	I	3.900
	II	7.800
	III	3.900
	IV	3.900
Dorsal Arm	II	3.900
	III	7.800
	IV	3,900
	VI	3,900
Ventral Arm	I	7,748
	III	2,600
	IV	3,900
	V	3,900
	Total	75,348

Due to the variability of the skin tone of different areas for the same subject, we encountered different FST labels for that person in different regions. It is clear from Table I that the FST scale III has the highest number of samples, which was found across all the locations. On the other hand, FST scale VI was only found in the dorsal arm, thus producing the smallest sample size based on FST labels. The numbers of samples across the gradient FST scale are as follows - FST I: 11,648, FST II: 11,700, FST III: 24,701, FST IV: 15599, FST V: 7,800, and FST VI: 3,900.

C. Classification using Convolutional Neural Network

To classify the skin tones of six categories (FST), we used a two-dimensional convolutional neural network (CNN) [18]. We applied CNN to four different locations individually, and combining all the locations, resulted in five different models for FST classification. CNN can capture Spatial and Temporal dependencies in the image through the application of relevant filters. It works best for complex images with dependencies of pixels throughout the image. The role of CNN is to reduce the pixel size of the original image in every hidden layer without losing critical features for a good prediction. CNN architecture consists mainly of two layers – the convolution layer and the pooling layer [18]. The CNN element involved in the convolution operation is called a kernel. The CNN architecture may have more than one convolutional layer. Conventionally, the first convolutional operation is responsible for low-level features. The model adapts with the addition of layers to capture the high-level features[18]. The final goal of this operation is to extract the high-level features from the input data. After this operation, the convolved feature is reduced. At the same time, the dimensionality is either increased or remains the same by padding operation.

The pooling layer is responsible for reducing the spatial size of the convolved feature. This reduces the computational power to process the data through dimensionality reduction. Also, it helps to find the most dominant features from the data. There are mainly two kinds of pooling - max pooling and average pooling. In this architecture, we used max pooling, which returns the maximum value from the portion of the image covered by the kernel.

The convolutional layer and the pooling layer together form a hidden layer of CNN. The number of hidden layers may be increased for capturing more meaningful features but with a more computational cost. After the process mentioned above is completed, the output of the last CNN layer is flattened and fed to a regular neural network for classification.

After flattening, it is fed to a fully connected layer which can learn the non-linear combinations of the features from the previous layer. The backpropagation algorithm is applied to every iteration of the training. After a number of epochs, the model can distinguish the important and unimportant features in the images and classify them using softmax classification techniques. In this research, the CNN architecture consists of four hidden layers followed by two dense layers, and finally, the softmax activation function determines the six classes of skin tones. The architecture of CNN is depicted in Fig. 2.

D. Hyperparameter tuning for CNN model

We applied three different techniques for hyperparameter tuning to acquire the best hyperparameters for classifying skin types. Three different techniques are - random search technique [19], hyperband technique [20], and Bayesian optimization technique [21]. All these three techniques were adopted from the Keras scikit learn package.

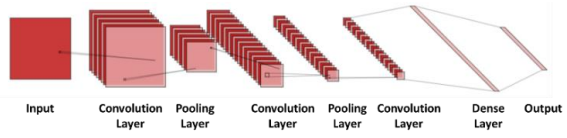

Fig. 2. **CNN Architecture.** The basic architecture of CNN consists of convolution layers and pooling layers followed by a dense layer. The number of layers may vary in different experiments. The image was generated using a publicly available tool (https://alexlenail.me/NN-SVG/LeNet.html)

Random search technique: This technique randomly samples from the list of hyperparameter search spaces. This algorithm has no end. So, the user needs to specify the number of trials. This method suffers the curse of dimensionality to reach a preset fixed sampling density. The drawback of this technique is there is no guarantee of finding the local minima.

Hyperband technique: It is a variation of random search but with the approach to find the best time allocation for each configuration. At first, the algorithm randomly selects hyper-parameter sets in the search space. Then it evaluates the validation loss after a chosen number of iterations and discards half of the lowest performers of the hyperparameter. The process is repeated till only one model is left. The drawback of this technique is that if the number of configurations is large, then some good configurations that are slow to converge at the beginning are discarded early. This means that good hyperparameters are lost.

Bayesian optimization technique: Bayesian optimization is built upon the Gaussian process regression, also known as kriging. It is efficient in tuning a few hyperparameters, but its efficiency degrades when the search space increases too much, up to a point where it is on par with random search. One major drawback is that it is not parallel, unlike random search. Gaussian processes suffer from a high computational cost if the number of evaluations is high. The regression task needs $O(n^3)$ operations. Thus, with a large number of samples, the time to run Bayesian optimization may take significant time.

In this work, we tuned ten hyperparameters, among which four hyperparameters are for the number of kernels in the four convolutional blocks, one for the dense unit, two for dropout values, one for dense activation, one for learning rate, and one for decay. All these hyperparameter values range in various numbers.

All the tuning algorithms were run for 20 trials and 100 epochs. The algorithms provide a set of best hyperparameters based on the maximum validation score. With the best set of hyperparameters, we ran the CNN pipeline again with a higher number of epochs (150). Later, the models were saved in the local machine, and the test sets were evaluated based on the models.

The dataset was split into the train and test sets according to the 80/20 ratio to avoid overfitting. The validation split in the CNN algorithm was chosen as 0.2, which means that 20 percent of the training set was used as the validation set. The validation set helps to determine how well the model has been trained by observing the validation loss. The training set was used to estimate the learning parameters, and the test set was used for performance evaluation.

E. Perfromance Evaluation of the CNN classifier

The classification performance was evaluated using two metrics: accuracy and f1-score. Accuracy measures how many positive and negative observations were correctly classified. F1-score combines precision and recall into one metric by calculating the harmonic mean between precision and recall. F1-score gives a better measure of the incorrectly classified cases than the accuracy metric. Accuracy is used when the true positives and true negatives are more important, while F1-score is used when the false negatives and false positives are crucial.

III. RESULT

A. Best hyperparameters from hyperparameter tuning

Three different techniques for hyperparameter tuning - random search, hyperband, and Bayesian optimization was applied to the dataset from four different locations individually and combined. Each of the techniques provided a set of best hyperparameter values. It implies that there is a total of 15 $(4\ locations \times 3\ techniques + combined\ location \times 3\ techniques)$ different sets of best hyperparameter values, among which the best hyperparameters were chosen based on the validation accuracy. Ten hyperparameter values were tuned to get the best output of the CNN models. The hyperparameters and their ranges are as follows:

a. Filters for convolutional layer 1: {32, 64, 128}

b. Filters for convolutional layer 2: {64, 128, 256}

c. Filters for convolutional layer 3: {128, 256, 512}

d. Filters for convolutional layer 4: {128, 256, 512}

e. Dropout 1: minimum value=0.0, maximum value=0.5, interval step=0.05

f. Dense unit: minimum value=32, maximum value=512, interval step=32

g. Dense activation functions: {relu, tanh, sigmoid}

h. Dropout 2: minimum value=0.0, maximum value=0.5, interval step=0.05

i. Learning rate: minimum value=0.0001, maximum value=0.01, log sampling was applied

j. Learning rate: minimum value=0.001, maximum value=0.1, log sampling was applied

The number of filters was increased with the increase of convolutional layers. The main idea here is that initially, CNN learns the low-level features with a few filters, and in the end, it learns the high-level features with a higher number of filters. It was noticed after the tuning that, even though the best hyperparameter values varied for different techniques, the validation accuracies were very high and close to each other. One such scenario, where data from all the locations were combined, is shown in Table II.

TABLE II. **BEST VALUES OF HYPERPARAMETERS FOR COMBINED DATASET.** THIS TABLE SHOWS THE BEST VALUES OF HYPERPARAMETERS FOR THREE DIFFERENT TECHNIQUES (HYPERBAND, RANDOM SEARCH AND BAYESIAN OPTIMIZATION). THE BEST SET WAS CHOSEN BASED ON THE VALIDATION ACCURACY.

HP Tuning	Conv-1	Conv-2	Conv-3	Conv-4	Dropout-1	Dense Unit	Dense Activation	Dropout-2	Learning rate	Decay	Score (Validation Accuracy)
Hyperband	12	64	512	512	0.05	384	sigmoid	0.35	0.000174	0.001123	79.52%
Random Search	128	128	512	512	0.3	352	sigmoid	0.05	0.000488	0.001234	82.25%
Bayesian Optimization	128	256	512	512	0	512	sigmoid	0	0.0001	0.001	84.50%

After getting the best hyperparameters, we ran the CNN algorithm again with these parameters, but we used a larger number of epochs (150). We ran CNN three times for all locations combined because of the three different sets of values coming from three different hyperparameter techniques used. This time we observed that the model converges and the training, validation, and test accuracies are high (> 98%), as shown in Fig 3. This figure indicates that we need the proper combination of hyperparameters to achieve high accuracy.

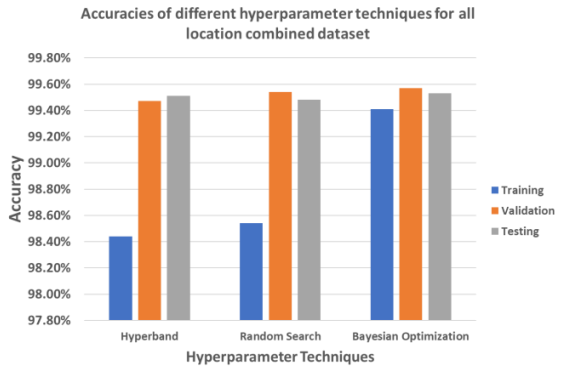

Fig. 3. **Accuracies of different hyperparameter techniques for combined dataset.** The figure indicates that different sets of best hyperparameters from three techniques produce high accuracies for the combined dataset.

Similar characteristics were also noticed in other dataset from different locations (dorsal hand, ventral hand, dorsal arm and ventral arm), which is not shown in the paper.

B. Evaluation Results

We evaluated the five different models using two evaluation metrics- accuracy and f1-score. Also, training loss and validation loss were observed after each model was trained. It was ensured that the loss was as minimum as possible. Also, the confusion matrix was analyzed using the test dataset to get a clear picture how the model works. Both model loss and confusion matrix for the combined test data is depicted in Fig. 4. We used the model with the hyperparameters from Bayesian optimization to generate the figure due to its slightly better test accuracy than the other two techniques. It is clear from the confusion matrix that the model can predict the true classes even though the dataset is imbalanced.

The same scenario was also noticed in all the other four different datasets from the four locations we used. The accuracies of the best model among the three techniques for the five different datasets are shown in Fig. 5. It is clear from Fig. 5 that with the best hyperparameter values, the CNN architecture was able to learn important features, and that is why the accuracies were higher for every dataset from four locations and combined locations as well.

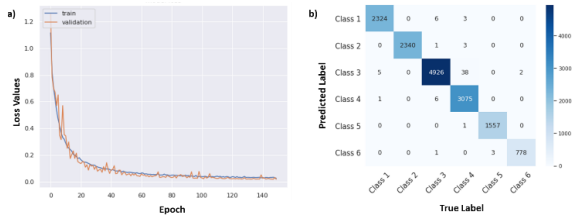

Fig. 4. **Model loss and confusion matrix for one model using the hyperparameters from bayesian optimization on combined dataset.** a) This is a loss vs epoch plot. We can see from the figure that with the increasing number of epochs the loss is going down. b) Confusion matrix for the test dataset. It is clear that most of the classes were correctly predicted..

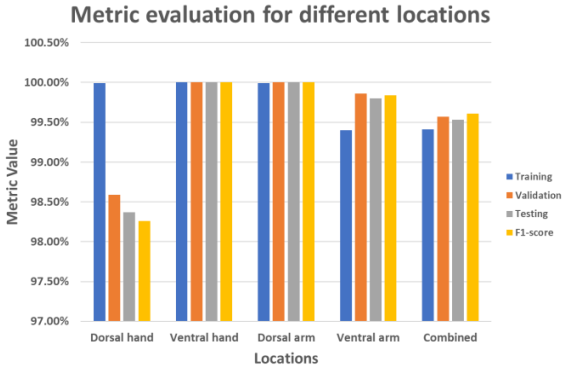

Fig. 5. **Metric values for different locations.** Metrics are: Training accuracy, validation accuracy, tesing accuracy, and F1-Score. The metric values are high for all the locations (> 98%).Ventral hand and dorsal arm have the highest values (~100%). On the other hand dorsal hand has the lowest validation, testing and f1-score which is still above 98%.

IV. Discussion

Our hypothesis was that with the proper hyperparameter tuning, we would be able to classify the skin tones of real human subjects with high accuracy. To test the hypothesis, first, we developed a dataset for FST skin tone types from four locations of five subjects, and they were labeled based on the gradient FST scaling. Next, we developed a deep learning framework optimized using three hyperparameter techniques.

From the data distribution, we noticed that the dataset is imbalanced. FST III has the highest number of samples, whereas FST VI has the lowest samples. Moreover, all the FST classes were not present in any of the four locations. For example, the dorsal hand had only three FST scaling samples: FST III, FST IV, and FST V. Also, from the metadata analyses, it was noticed that due to skin pigmentation variability, different locations of the same subject have different FST skin types.

The result shows that the classification accuracy and f1-score are high for all locations. Also, when we combined the dataset, the accuracy and f1-score did not deteriorate. In all the cases, the score was more than 98%. But this may happen due to overfitting of the data. We used 65 frames of a single video, and these 65 frames came from the same location. All the frames for the single videos may bear the same information, which may cause the testing data to be the same as the training data.

V. Conclusion and Future Remarks

This study represents a preliminary experiment and observation of skin tone classification for in-vivo subjects, which would help in our future study to incorporate the effect of skin tone in physiological wound analysis.

Future studies will incorporate a large number of subjects' images to develop an extensive benchmark dataset. The larger sample size will help the machine learning model learn more confidently for more reliable results prior to implementing it as a correction factor during physiological imaging using the smartphone-based optical device (SPOT).

Acknowledgment

This work was partially supported by the NSF CAREER Award IIS 1901628 to AMM; NIH R01EB033413 to AG and AMM; Coulter Bridge Grant to AG and AMM. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

[1] S. R. Nussbaum et al., "An Economic Evaluation of the Impact, Cost, and Medicare Policy Implications of Chronic Nonhealing Wounds," Value Health, vol. 21, no. 1, pp. 27–32, Jan. 2018.

[2] M. Neidrauer, L. Zubkov, M. S. Weingarten, K. Pourrezaei, and E. S. Papazoglou, "Near Infrared Wound Monitor Helps Clinical Assessment of Diabetic Foot Ulcers," https://doi.org/10.1177/193229681000400404, vol. 4, no. 4, pp. 792–798, Jul. 2010.

[3] G. Mulder, M. Tenenhaus, and G. F. D'Souza, "Reduction of diabetic foot ulcer healing times through use of advanced treatment modalities," Int. J. Low. Extrem. Wounds, vol. 13, no. 4, pp. 335–346, 2014.

[4] D. W. Paul et al., "Noninvasive imaging technologies for cutaneous wound assessment: A review," Wound Repair Regen., vol. 23, no. 2, pp. 149–162, 2015.

[5] K. Kacie, C. Fernandez, and A. Godavarty, "Tissue Oxygenation Measurements using a Non-Contact, Smartphone-based Near-Infrared Optical Device," Biophotonics Congr. Biomed. Opt. 2020, 2020.

[6] K. Kaile, C. Fernandez, and A. Godavarty, "Development of a Smartphone-Based Optical Device to Measure Hemoglobin Concentration Changes for Remote Monitoring of Wounds," Biosens. 2021, Vol. 11, Page 165, vol. 11, no. 6, p. 165, May 2021.

[7] K. Kaile et al., "Low-cost smartphone based imaging device to detect subsurface tissue oxygenation of wounds," Proc. SPIE 10869, Opt. Biophotonics Low-Resource Settings V, 2019.

[8] K. Kaile and A. Godavarty, "Development and Validation of a Smartphone-Based Near-Infrared Optical Imaging Device to Measure Physiological Changes In-Vivo," Micromachines 2019, Vol. 10, Page 180, vol. 10, no. 3, p. 180, Mar. 2019.

[9] M. Sobhan, K. Kalie, A. Al Mamun, A. Godavarty, and A. M. Mondal, "Skin Tone Benchmark Dataset for

Diabetic Foot Ulcers and Machine Learning to Discover the Salient Features," Int. Conf. Image Process. Comput. Vision, Pattern Recognit., 2022, Accessed: Sep. 29, 2022. [Online]. Available: https://www.researchgate.net/publication/362517589.

[10] K. Kaile, M. Sobhan, A. Mondal, and A. Godavarty, "Machine learning algorithms to classify Fitzpatrick skin types during tissue oxygenation mapping," Opt. InfoBase Conf. Pap., pp. 10–11, 2022.

[11] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," J. R. Stat. Soc. B, vol. 73, pp. 273–282, 2011.

[12] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for Multi-Cluster data," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., pp. 333–342, 2010.

[13] D. M. Reif, A. A. Motsinger, B. A. McKinney, J. E. Crowe, and J. H. Moore, "Feature selection using a random forests classifier for the integrated analysis of multiple data types," Proc. 2006 IEEE Symp. Comput. Intell. Bioinforma. Comput. Biol. CIBCB'06, pp. 171–178, 2006.

[14] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines," Mach. Learn. 2002 461, vol. 46, no. 1, pp. 389–422, 2002.

[15] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "L2,1-Norm Regularized Discriminative Feature Selection for Unsupervised Learning."

[16] A. Al Mamun et al., "Multi-Run Concrete Autoencoder to Identify Prognostic lncRNAs for 12 Cancers," Int. J. Mol. Sci. 2021, Vol. 22, Page 11919, vol. 22, no. 21, p. 11919, Nov. 2021.

[17] C. Saravanan, "Color image to grayscale image conversion," 2010 2nd Int. Conf. Comput. Eng. Appl. ICCEA 2010, vol. 2, no. January, pp. 196–199, 2010.

[18] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," Nov. 2015.

[19] J. Bergstra, J. B. Ca, and Y. B. Ca, "Random Search for Hyper-Parameter Optimization Yoshua Bengio," J. Mach. Learn. Res., vol. 13, pp. 281–305, 2012, Accessed: Sep. 29, 2022. [Online]. Available: http://scikit-learn.sourceforge.net.

[20] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization," J. Mach. Learn. Res., vol. 18, pp. 1–52, Mar. 2016.

[21] R. Turner et al., "Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020," Apr. 2021.