BMI estimation from facial images using residual regression model

Quoc-Trung Pham
Computer Science Department
Vietnamese-German University
Binhduong, Vietnam
cs2015_trung.pq@student.vgu.edu.vn

Anh-Tuan Luu
Finance Devision
Prudential Vietnam
Hochiminh, Vietnam
luu.anh.tuan@prudential.com.vn

Thanh-Hai Tran
School of Electronics and Telecommunications
Hanoi University of Science and Technology
Hanoi, Vietnam
hai.tranthithanh1@hust.edu.vn

***Abstract*—Body Mass Index (BMI) has the potential to disclose a variety of health and lifestyle concerns. Predicting BMI from facial images is an interesting but challenging problem in computer vision. Previous works focus mainly on feature extraction step of the whole BMI estimation process. Little attention has been paid to the regression module. In this paper, we propose a new architecture for the regression module which composes of multiple blocks. Each block has several sub-blocks composing of dense layer, batch-normalization, activation, dropout. In addition, we take advantage of the residual principle from ResNet by adding residual connections in the regression blocks. We integrate the proposed regression model just after the state-of-the-art feature extractor ResNet and train the network in an end-to-end manner. Extensive experiments on the VIP_Attributes dataset show that thanks to the new residual regression model, the estimation error reduces up to 22% in comparison to the original method.**

***Index Terms*—BMI prediction, residual regression model, facial images, deep learning**

I. Introduction

Face photos provide a wealth of biometric data, including identification, gender, age, weight, and body mass index (BMI) [1]. The use of machine learning-based algorithms to decode facial signals has attracted the interest of computer scientists. BMI (given by $\frac{weight(kg)}{height^2(m)}$) is commonly used as a body fat indicator in monitoring one's own health and medical research [2]. It has the potential to disclose a variety of health and lifestyle concerns. There are strong links between BMI and certain illnesses, such as malignancies, unstable angina, diabetes, to name a few.

The ability to automatically estimate BMI from face photos is quite useful for healthcare monitoring, researching obesity in huge populations [3]. According to previous findings, face adiposity is linked to subjective health and is crucial for predicting BMI [1], [3]–[6].

However, estimating BMI from facial photos is a difficult task [6]. The big problems can be mentioned such as lack of data for training and testing models, uneven distribution of data [5]. In addition, there are also issues such as variation in lighting conditions, shooting angle, resolution, etc. In the past, BMI value was calculated using weight and height and was done manually by using high-quality pieces of equipment. However, when physical measures are unavailable, self-reported measures are frequently employed as a backup measure. Participants tend to overestimate their height and underestimate their weight, resulting in erroneous BMI estimates [4].

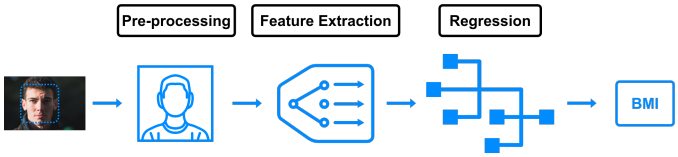

Fig. 1. BMI estimation process diagram

Image pre-processing, facial features extraction, and final regression are main components in a typical process of estimating BMI from two-dimensional (2D) facial photographs [3]. An overview diagram for the whole process is in Fig. 1. Recent research is focusing mainly on the second part of the process [2], [3], [5]. Only a few research have investigated architectures of the final regression model. Currently, to the best of our knowledge, only some research papers use deep learning models for the prediction step. However, those models seems to be quite simple, including only a couple of fully connected layers.

For instance, in [6], the authors used a model consisting of one layer with a single node. A regression module consists of three dense layers with the number of nodes 512, 32, 1 respectively, interspersed between dense layers are drop out layers, activation layers to avoid model over-fitting was used in [7]. In [8], the authors proposed an end-to-end model, with the regression module made up of two dense layers with dimensions of 200 and 1, respectively. Similarly, the researchers proposed a simple model with three dense layers with the number of nodes being 256, 64, 1 [9]. Simple regression architecture can not capture high correlation of data then the estimation performance can be limited.

With that motivation, we investiage deeper architecture for regression model. Specifically, we propose a new architecture which composes of multiple blocks. Each block has several sub-blocks composing of dense layer, batch-normalization, activation, dropout. In addition, we take the advantage of residual principle of ResNet and add residual connections in

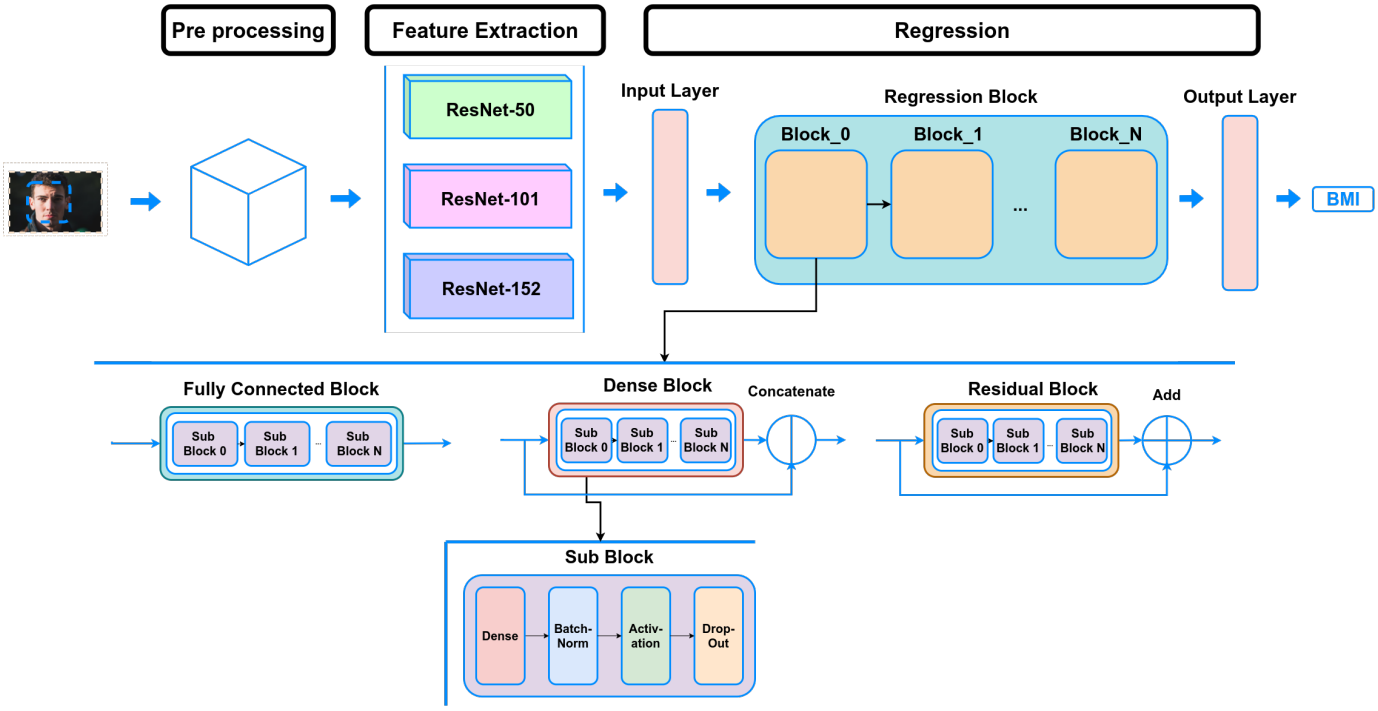

Fig. 2. The regression module

the regression blocks. We integrate the proposed regression model just after the state-of-the-art feature extractor ResNet and train the network in an end-to-end manner.

The remainder of the paper is organized as follows. In Section II we review work on the existing methods used for the prediction of BMI. Section III introduces the out proposed face-based BMI estimation algorithm to improve the current result. Datasets and experimental protocol are described in Section IV. Section V we present experiments validating the effectiveness of the proposed method and a discussion thereof. Finally, the conclusions are given in Section VI.

II. Related works

In literature, there exist different methods proposed to estimate BMI directly from 2D images of the face. Basically these methods can be divided into the following main groups: geometry features based methods and deep-learning features based methods.

Methods belonging to the first group used some indexes based on scientific metrics such as Cheek-to-jaw-width (CHWR), Width-to-upper-facial-height (WHR), etc., [10], [11]. These features are extracted directly by calculating the key points on the face. They are then used to estimate BMI through a machine learning algorithm such as Support Vector Regression (SVR) in the form of regression [11], [12].

Methods belonging to the second group are widely used in recent scientific studies. They use a pre-trained deep learning model to extract features directly from the facial image. These features are then used to predict BMI, through a machine learning algorithm.

Some works use directly extracted features for classification sample as underweight, normal, overweight, obese, or even severely obese [13]–[15] by using typical machine learning algorithms [3], [8], [16], [17]. Other works integrate a regression model in an end-to-end deep learning framework to predict a BMI value. However, the regression model seems to be still simple thus can not learn the hidden correlation between samples [6].

III. Proposed methods

A. General framework

In this paper, we propose a framework that deploys ResNet as a features extractor and a new regression module as the predictor. Our framework can be trained end-to-end. Figure 2 shows the main components of our framework. Our framework is inspired from the original method [6] that uses ResNet as feature extractor. The difference lies in the regression module. In the following sections, we will review the principle of ResNet and various ResNet architectures that will be used for investigating our framework. We then describe in more detail our proposed residual regression blocks.

B. ResNet as feature extractor

Most of the current research shows that the pre-trained model of ResNet gives better results in comparison with other deep models [6]–[8]. Therefore, we use ResNet's pre-trained models for feature extraction. To ensure that our module shall not over-fit, three different architectures of ResNet shall be

investigated to extract the feature vector of a given facial image.

As explained in the original work that introduced ResNet [18], it is difficult to train deep neural networks due to the problem of vanishing or exploding gradients. When deep neural networks converge, other issues emerge such as accuracy saturation and fast degradation. This issue can be resolved by using the residual blocks [18] (Fig. 3).

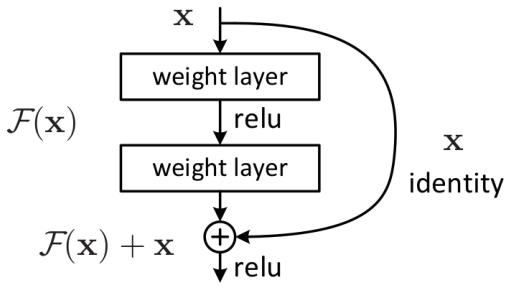

Fig. 3. Residual block of ResNet [18]

Based on this idea, ResNet's authors proposed different architectures. Among them, ResNet 50, ResNet 101, and ResNet 152 give the best results, in which each residual block has 3 layers. The ResNet-50 is constructed by combining many 3-layer bottleneck blocks, yielding a 50-layer neural network. That is the reason why they call it ResNet-50. There are 3.8 billion FLOPs in this model. With the same idea, ResNets with 101 and 152 layers are developed by using more 3-layer blocks [18].

C. Residual reegression model

As aforementionned in introduction section, the regression model in the orignal paper [6] is quite simple. It cannot learn the correlation of data samples. Therefore, we propose a more sophisticated regression model which inspires the same idea of ResNet with residual blocks.

Our proposed regression module includes several blocks. The number of blocks is selected in the range from one to six. Each block composes one or more sub-blocks. Each sub-block has 4 layers. To avoid the problem of overfitting raised by a complex model, we add a group of other layers like batch normalization, activation, dropout in each sub-block after each dense layer. Dense layer size in turn will be selected among different values. For a completed value of configuration space please refer to Tab. I.

TABLE I
TESTING CONFIGURATION SPACE FOR REGRESSION MODULE

Option	Values
Pre-trained Network	ResNet-50 ResNet-101 ResNet-152
Type of module	Fully-Connected Residual Dense
Num of blocks	1, 2, 3, 4, 5, 6
Num of sub-blocks	1, 2, 3, 4, 5, 6
Dense size	256, 512, 1024, 2048, 3072, 4096
Drop out rate	0.0, 0.1, 0.2, 0.3 0.4, 0.5, 0.6, 0.7
Activation	RELU [19], Tanh SELU [20], ELU [21]
Loss	LogCosh, MAE, MAPE [22] MSE, MSLE, Smooth L1 [23]
Optimizer	AdaDelta [24], AdaGrad [25] Adam [26], SGDW [27] AdamW [27], Ftrl [28] Nadam [29], RMSprop [30]

IV. EXPERIMENTS

A. Dataset

In this research, we evaluate our proposed framework on a benchmark dataset VIP [6]. The images and BMI are available in the authors website: http://www.antitza.com/VIP_Attribute-dataset.html. The dataset consists of 1026 subjects (mostly celebrities) collected from different websites, with equal numbers of men and women. The images are mostly frontal images however with the presence of makeup, plastic surgery, beards, or even effects from photo editing software. Example images of subjects can be found in Fig. 4.

For women, the mean of BMI value (μ) is 20.87, the standard deviation (σ) is 3.71, while for men μ is 25.21, and σ is 3.57 [6]. Normally, BMI value can be classified as one of 4 groups: under-weight, healthy, over-weight, obesity, the number of samples in each group can be found in Tab. II.

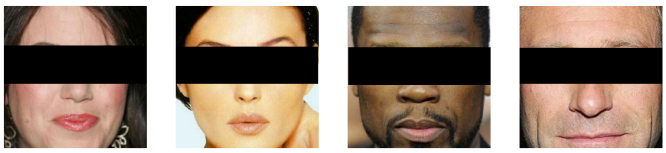

Fig. 4. Sample image from VIP dataset.

TABLE II
NUMBER OF SUBJECTS IN EACH BMI CLASSES OF VIP_ATTRIBUTE DATASET

Class	Range [2]	Number of subjects
Under Weight	BMI $<$ 18.5	77
Healthy	18.5 $\leq$ BMI $<$ 25	706
Over-weight	25 $\leq$ BMI $<$ 30	196
Obesity	BMI $>$ 30	47

Besides, to evaluate the model robustness, we use our model trained on VIP_Attribute dataset to evaluate on a small sample dataset that we collected from a website https://wiki.d-addicts.com. With this dataset, we want to assess the performance of the models on 4 classes of BMI. The dataset contains 64 subjects, each belongs to one of the following groups: under-weight, healthy, over-weight, obesity. Each group has 16 subjects.

B. Model training and testing

For training the model, the dataset is randomly split into train-set (800 samples), test-set (426 samples) as [5] for a fair comparison. To ensure that the model results are independent of a certain split of train-test, repeated k-fold cross-validation is used. Accordingly, the dataset will be divided into 5 folds, 4 folds will be used as train data, the remaining fold will be used to test the model. In addition, we will repeat this process 5 times to make sure the data is randomly divided. In simple terms, each model will be trained and tested 5 folds $\times$ 5 repeats = 25 times.

The model shall be tested on the test set with 2 returned results: Mean Absolute Error (MAE) and Smooth L1 Loss [23]. However, we will focus mainly on the MAE comparison because this value is used to compare the performance of estimators. In the training process, the model shall be trained up to 500 epochs.

After investigating different parameters, we achieve the structure with a detailed configuration. For the pre-trained model, ResNet-50 gives the best results, which also coincides with recent research in [7]. At the same time, the optimal architecture for regression module can be found in Tab. III.

TABLE III
BEST CONFIGURATION FOR REGRESSION MODULE

Configuration	**Our method**	**Method** [6]
Pre-trained model	ResNet-50	ResNet-50
Type of block	Residual	1 Dense layer only
Number of blocks	2	n.a
Number of sub-blocks	3	n.a
Dense layer linear size	2048	1
DropOut rate	0.5	n.a
Activation function	RELU	n.a
Optimizer	Adam	SGDW
Loss function	SmoothL1	SmoothL1

C. Experimental results

1) Results on VIP dataset: To evaluate the performance of our proposed regression model, we compare our framework with the work in [6] on the VIP_Attribute dataset. After finding the optimal structure (Tab. III), we compare the efficiency of this structure with the original structure [6] and existing works using MAE value.

As the code implementing the original model in [6] was not provided, as well as the information of data split for training and testing was not clearly defined in the paper. Therefore we re-implement the model, then train and test it using our described splitting above with the same training parameters as indicated in the paper [6]. In the following comparision, we will compare MAE obtained by our method with with MAE: one reported in [6] and one produced from our re-implementation of [6].

As shown in Tab. IV, all the mean, and standard deviation values of the new structure are lower than those of the original structure. Specifically, our μ value is $\approx 22\%$ lower (2.14 vs 2.76), the σ is $\approx 43\%$ lower (0.17 vs 0.3). Besides, it can

TABLE IV
CROSS VALIDATION TEST RESULT WITH 5-FOLDS ON VIP_ATTRIBUTE DATASET

Metric	**Our method**	**Method [6]*[1]**	**Method [6][2]**
Number of trials	25	25	n.a
Max MAE	2.58	3.3	n.a
Min MAE	1.83	2.1	n.a
Average MAE(μ)	2.14	2.76	2.3
Std. MAE(σ)	0.17	0.3	+ 0.06
25% MAEs	1.99	2.56	n.a
50% MAEs	2.11	2.79	n.a
75% MAEs	2.28	2.91	n.a

[1]The re-implemented model [6]
[2]Test result reported in [6].

be seen that the 75% MAE value of our module less than or equal to the value of 2.28, which is even smaller than the 25% percentile (2.56) of the current module. In other words, 75% MAE values of the proposed method are smaller than 75% MAE values of the current one.

As the data is not quite large (only 1026 samples) and heavily bias, with most of subjects are in either healthy or over-weight groups as in Tab. II. Therefore, the results of the models are very sensitive to each data division. Even in that case, our proposed structure has a narrower difference between min and max value, which 0.75 in comparison to 1.2 ($\approx 16\%$).

Distributions of MAE from both modules (our proposed regression model and the original regression model [6]) can be seen in Fig. 5. As the value of σ is significantly smaller, most of the MAE values of our model lie close to the expected value. Thus, it can be said that our regression model allows estimating BMI much only better but also more stable in comparison with the original regression model on this dataset.

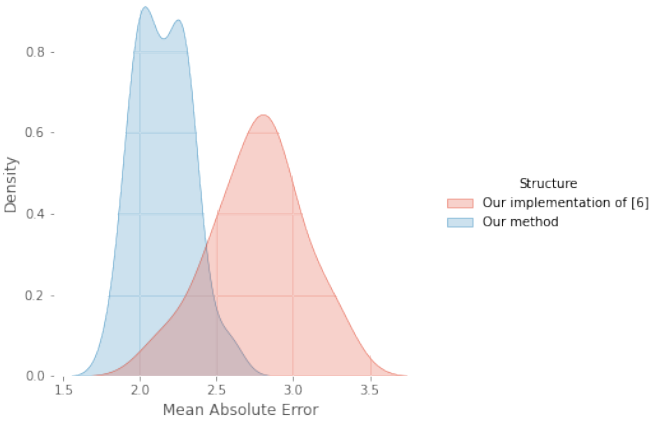

Fig. 5. MAE distribution of our model and the original model [6] using Repeated 5 folds cross-validation.

We also provide the a best result of our model and current model on a particular train/test split in Fig. 6.

A comparison of our method with current results on MAE

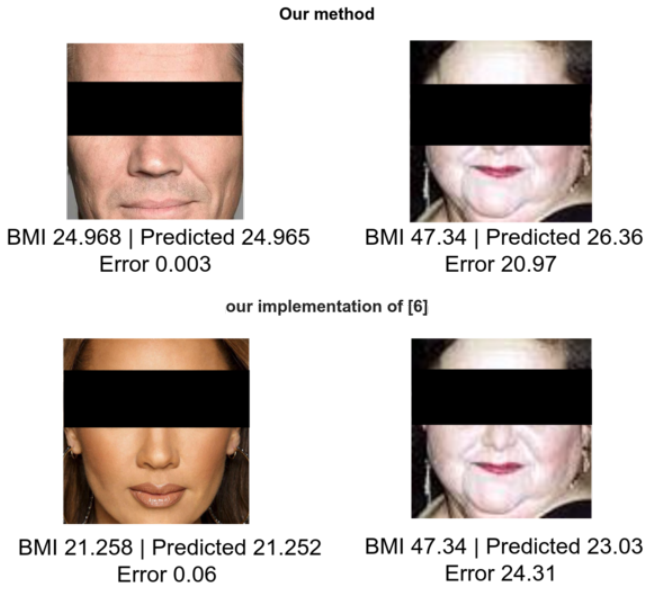

Fig. 6. A sample of best and worst prediction of the models on VIP_Attribute dataset.

on VIP dataset from other researches can be found in Tab. V. In the best case, we can achieve a quite good result in comparison with other methods. Reg-GAP methods can provide a better result. In addition to using the feature vector directly extracted, it requires regional information extracted by using semantic segmentation. However, this makes the model more complex, and the calculation process also requires more computing power and even takes more time.

TABLE V
MAE ON VIP DATASET WITH CURRENT RESEARCH

Model	Method	MAE
Our best	Deep Learning	1.83
Reported in [6]	Deep learning	2.3
Reg-GAP [7]	Deep Learning	1.73
LD_CCA [5]	Machine learning	2.23

2) Results on our dataset: Finally, the performance of our proposed model on the small dataset built by ourselves is described in Tab. VI. Evaluation results on the new dataset show that our proposed module gives the best results for the healthy group, reaching 1.48 in MAE, and the worst for the obesity group, reaching 10.11 in MAE.

This result completely matches the number of subjects in each group of VIP_Attribute dataset as shown in Tab. II, with the number of samples in the healthy group accounting for the largest number 706/1026, and the obesity group accounting for at least 47/1026. Despite reducing the error, our method still cannot overcome the data bias.

In comparison with the original method [6], our proposed model has a better result on 3 out of 4 classes. The original model [6] is heavily affected by the bias in the training dataset. The best result of our model and origial model [6] can be found in Fig. 7.

TABLE VI
MEAN ABSOLUTE ERROR ON DIFFERENT BMI CLASSES OF OUR SAMPLE DATASET

Class	Number of subjects	Our method	Our implementation[1]
Under-weight	16	4.78	3.1
Healthy	16	1.48	1.76
Over-weight	16	2.84	4.31
Obesity	16	10.11	10.97

[1]The re-implemented model [6]

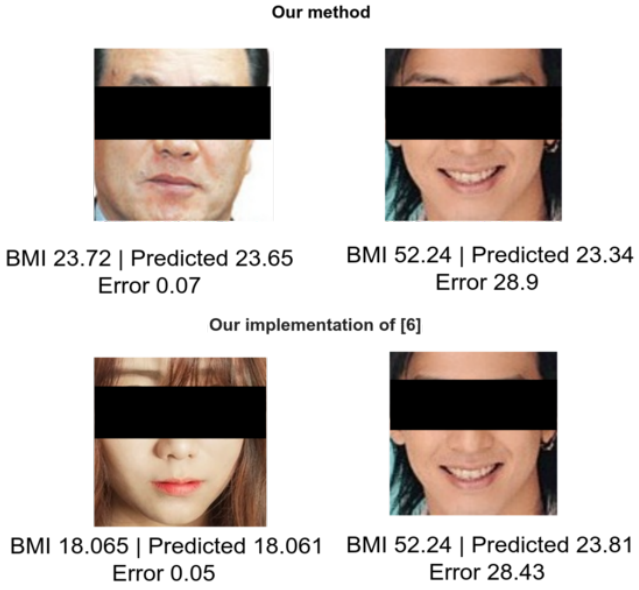

Fig. 7. Best and worst prediction of models on our dataset.

V. CONCLUSION

In this paper, we have presented a new regression model for BMI estimation from facial image. Different from existing methods, our proposed regression model is deeper. It contains multiple blocks, each block consists of multiple sub-blocks with different layers such as dense layer, batch normalization, activation and drop-out. We also inspired the idea of ResNet to add a residual connection in our regression model. Our proposed method takes a facial image as input, extracts features using ResNet and predicts a BMI value using the proposed regression model. We integrated our proposed regression model with ResNet feature extractor in a unified framework and trained it in end-to-end manner. Thanks to the new regression model, we achieved better results comparing to the original model on the same dataset on VIP_attibute dataset. Moreover, the proposed regression model produces more stable result. In future works, we will evaluate the

proposed model on other dataset, combine facial landmark features to improve estimation accuracy.

REFERENCES

[1] L. Wen and G. Guo, "A computational approach to body mass index prediction from face images," *Image and Vision Computing*, vol. 31, no. 5, p. 392–400, 2013.

[2] F. Q. Nuttall, "Body mass index," *Nutrition Today*, vol. 50, no. 3, p. 117–128, 2015.

[3] M. Jiang, Y. Shang, and G. Guo, "On visual bmi analysis from facial images," *Image and Vision Computing*, vol. 89, p. 183–196, 2019.

[4] M. Barr, G. Guo, S. Colby, and M. Olfert, "Detecting body mass index from a facial photograph in lifestyle intervention," *Technologies*, vol. 6, no. 3, p. 83, 2018.

[5] M. Jiang, G. Guo, and G. Mu, "Visual bmi estimation from face images using a label distribution based method," *Computer Vision and Image Understanding*, vol. 197-198, p. 102985, 2020.

[6] A. Dantcheva, F. Bremond, and P. Bilinski, "Show me your face and i will tell you your height, weight and body mass index," *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018.

[7] N. Yousaf, S. Hussein, and W. Sultani, "Estimation of bmi from facial images using semantic segmentation based region-aware pooling," *Computers in Biology and Medicine*, vol. 133, p. 104392, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482521001864

[8] H. Siddiqui, A. Rattani, D. R. Kisku, and T. Dean, "Al-based bmi inference from facial images: An application to weight monitoring," in *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec 2020, pp. 1101–1105.

[9] A. Haritosh, A. Gupta, E. S. Chahal, A. Misra, and S. Chandra, "A novel method to estimate height, weight and body mass index from face images," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*, Aug 2019, pp. 1–6.

[10] C. Mayer, S. Windhager, K. Schaefer, and P. Mitteroecker, "Bmi and whr are reflected in female facial shape and texture: A geometric morphometric image analysis," *Plos One*, vol. 12, no. 1, 2017.

[11] D. D. Pham, J.-H. Do, B. Ku, H. J. Lee, H. Kim, and J. Y. Kim, "Body mass index and facial cues in sasang typology for young and elderly persons," *Evidence-Based Complementary and Alternative Medicine*, vol. 2011, p. 1–9, 2011.

[12] B. Scholkopf, A. Smola, R. Williamson, and P. Bartlett, "New support vector algorithms," *Neural computation*, vol. 12, pp. 1207–45, 06 2000.

[13] L. F. Polania, G. M. Fung, and D. Wang, "Ordinal regression using noisy pairwise comparisons for body mass index range estimation," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019, pp. 782–790.

[14] C. Y. Fook, L. C. Chin, V. Vijean, L. W. Teen, H. Ali, and A. S. A. Nasir, "Investigation on body mass index prediction from face images," in *2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, March 2021, pp. 543–548.

[15] B. J. Lee, J.-H. Do, and J. Y. Kim, "A classification method of normal and overweight females based on facial features for automated medical applications," *Journal of Biomedicine and Biotechnology*, vol. 2012, p. 1–9, 2012.

[16] E. Kocabey, M. Camurcu, F. Ofli, Y. Aytar, J. Marin, A. Torralba, and I. Weber, "Face-to-bmi: Using computer vision to infer body mass index on social media," *CoRR*, vol. abs/1703.03156, 2017.

[17] K. Wolffhechel, A. C. Hahn, H. Jarmer, C. I. Fisher, B. C. Jones, and L. M. Debruine, "Testing the utility of a data-driven approach for assessing bmi from face images," *Plos One*, vol. 10, no. 10, 2015.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[19] A. F. Agarap, "Deep learning using rectified linear units (relu)," *CoRR*, vol. abs/1803.08375, 2018. [Online]. Available: http://arxiv.org/abs/1803.08375

[20] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," *CoRR*, vol. abs/1706.02515, 2017. [Online]. Available: http://arxiv.org/abs/1706.02515

[21] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," 2015.

[22] A. de Myttenaere, B. Golden, B. Le Grand, and F. Rossi, "Mean absolute percentage error for regression models," *Neurocomputing*, vol. 192, p. 38–48, Jun 2016. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2015.12.114

[23] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964. [Online]. Available: http://www.jstor.org/stable/2238020

[24] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://arxiv.org/abs/1212.5701

[25] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 07 2011.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[27] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," *CoRR*, vol. abs/1711.05101, 2017. [Online]. Available: http://arxiv.org/abs/1711.05101

[28] H. B. Mcmahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, D. Golovin, and et al., "Ad click prediction," *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013.

[29] T. Dozat, "Incorporating nesterov momentum into," 2015.

[30] T. Tieleman and G. Hinton, "Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude," COURSERA: Neural Networks for Machine Learning, 2012.