International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:13, No:7, 2019

Foot Recognition Using Deep Learning for Knee Rehabilitation

Rakkrit Duangsoithong, Jermphiphut Jaruenpunyasak, Alba Garcia

***Abstract*—**The use of foot recognition can be applied in many medical fields such as the gait pattern analysis and the knee exercises of patients in rehabilitation. Generally, a camera-based foot recognition system is intended to capture a patient image in a controlled room and background to recognize the foot in the limited views. However, this system can be inconvenient to monitor the knee exercises at home. In order to overcome these problems, this paper proposes to use the deep learning method using Convolutional Neural Networks (CNNs) for foot recognition. The results are compared with the traditional classification method using LBP and HOG features with kNN and SVM classifiers. According to the results, deep learning method provides better accuracy but with higher complexity to recognize the foot images from online databases than the traditional classification method.

***Keywords*—**Convolutional neural networks, deep learning, foot recognition, knee rehabilitation.

I. Introduction

GENERALLY, knee physical rehabilitation [1] aims to increase the strength, mobility and fitness of the joint for a patient. This exercise is observed by a physical therapist (PT) to provide feedback to the patient, such as the limited range of motion of knee (angle). Normally, the PT uses a goniometer to measure angle of the knee joint during physical rehabilitation. However, it may be not suitable for home rehabilitation or many patients in hospital. Normally, there are many examples of knee rehabilitation such as using body weight, with machines and free weights [2]. Considering the main postures of this rehabilitation, the patient's foot is the main moving part of the body with the direction under the instructions from PT, while the knee joint position of the patient should be fixed to protect from the pain that might be occurred. The angle of the knee can be measured by tracking the movement of the foot during knee rehabilitation.

Using a markerless caption system (MCS) [3] for tracking a foot with camera can monitor a patient during the rehabilitation program. It is low cost and easy to setup, however, there are still some detection and tracking problems such as an occlusion from other objects, the light condition of the room and the complex background that are difficult to recognize the foot images. Some patients might prefer to wear shoes or socks than to be a barefoot when doing a rehabilitation program which is difficult to detect the foot location in the image.

In computer vision application, it becomes useful to identify objects in raw images, search items and translation information. The general model of machine learning is supervised learning which consists of a proper data set of images that are labelled with its classes. Generally, machine learning techniques require a feature generation process to convert the low-level information such as the pixel values to suitable feature vectors such as the local binary patterns (LBP) [4] and the histogram of oriented gradients (HOG) [5], [6]. These features are able to represent its class by using a classifier which is trained from the labelled data set. Nevertheless, if the data set is a variety in the same class, the feature vectors might not be satisfied to fit with the classifier.

Deep learning [7] is a new technique for machine learning. It can additionally apply to the low level and high level information in a large data set. The deep learning architecture is normally similar to artificial neural networks but it has more hidden layers and a greater number of nodes which enable to identify an object with high accuracy such as handwritten digits, pedestrian and object detection, etc. This paper proposed the foot recognition using CNNs and compared it with traditional classification methods.

II. Method

A. Traditional Classification Methods for Foot Recognition

As can be seen from Fig. 1, there are four main steps of foot recognition.

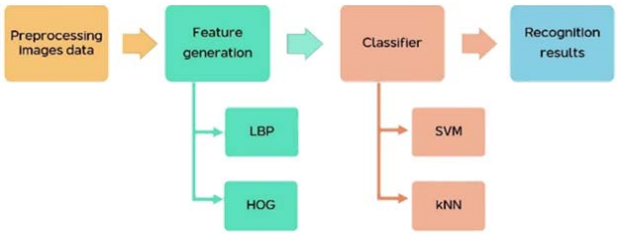

Fig. 1 Diagram of foot recognition

1. Preprocessing Images Data

The raw images are normally labelled, cropped and converted into the suitable color space. This data set consists of two types of data; positive and negative images.

Rakkrit Duangsoithong is with the Department of Electrical Engineering, Faculty of Engineering Prince of Songkla University, Hat Yai, Songkhla, 90112 Thailand (corresponding author; e-mail: rakkrit.d@psu.ac.th).
Jermphiphut Jaruenpunyasak is with the Institute of Biomedical Engineering, Faculty of Medicine Prince of Songkla University, Hat Yai, Songkhla, 90112, Thailand (e-mail: jjermphi@medicine.psu.ac.th).
Alba Garcia is with the School of Computer Science and Electronic Engineering Wivenhoe Park Colchester CO4 3SQ, University of Essex, United Kingdom (e-mail: alba.garcia@essex.ac.uk).

2. Feature Generation

This process creates the feature vectors from a raw image. The feature vectors represent the information of its image. In this research, the LBP and HOG features are used as the feature vectors in the experiment.

- **Local Binary Patterns (LBP)** [4], [8] is an image feature that uses the LBP mask to calculate the gray images, as demonstrated in Fig. 2. This mask generally divides the images into cells determined by a radius of its mask. For example, if the radius is one, the neighbor's pixel around the cells is 8 points, as shown in Fig. 3. The output of the LBP mask is normalized by using a histogram of its LBP values and the dimension reduced by the rotation invariant uniform LBP.

Fig. 2 Diagram of LBP calculation

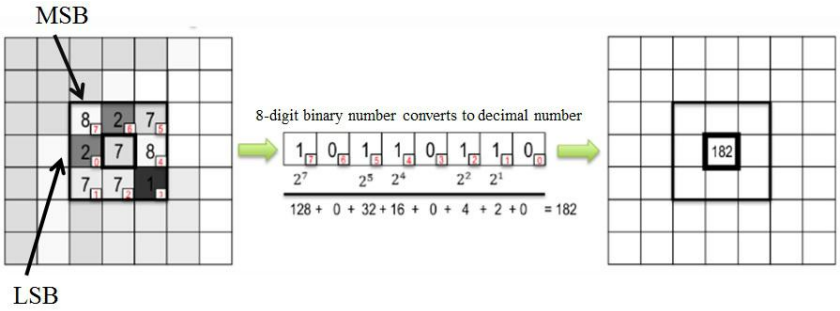

Fig. 3 Example of the LBP pixels calculation along a clockwise

- **Histogram of Oriented Gradients (HOG)** [5], [6], [9] is another image feature that uses the histograms of oriented gradients of a gray image. Generally, its process initializes the parameters such as the sizes of cells, blocks and bins. The image calculates the gradients of the horizontal axis, vertical axis and the magnitude angle of gradients. This edge histogram is created as a gradient vote depending on the bin parameters with a normalized magnitude of gradient vote. Finally, the 2-dimensional features are changed into a single vector of features to represent the information of the image, as illustrated in Fig. 4.

Fig. 4 Diagram of HOG calculation

3. Classification

Basically, the feature vectors of an image are trained with the labelled class (supervised learning). The classifier model will learn from both negative and positive images. In this research, SVM and kNN are used to evaluate the accuracy of the model.

- **Support Vector Machine (SVM)** [10] is a well-known classifier that use the hyperplane to find the maximum margin between the classes. To identify the suitable hyperplane, it requires the maximization of the distances between the nearest item point of a class and the decision plane. Its distance is also referred to as a margin. As a result, the selection of plane with a high margin will provide a robustness classifier.
- **k Nearest Neighbor (kNN)** is another well-known classifier [11]. It is also known as a lazy algorithm because it does not need a training phase. A distance between features and classes such as Euclidean distance is calculated. The final class will be calculated using majority vote from k samples that have the smallest distance between the feature and class.

4. Foot Recognition Results

The last step of the proposed system is to evaluate the performance of foot recognition from the test images. This paper uses sensitivity and specificity to evaluate the performance of the system.

B. CNN Methods for foot Recognition

Among the deep learning algorithms, CNNs have continuously been the effective method in image recognition, object tracking, and face analysis. In general, CNNs consist of varying component elements of three main layers, as shown in Fig. 5.

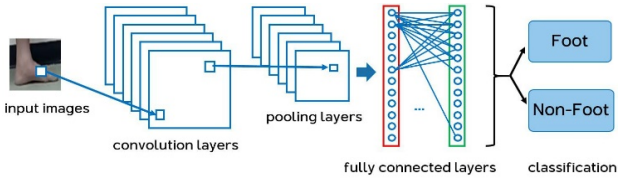

Fig. 5 Diagram of CNNs for foot recognition

- **Convolutional Layers** are the main core of filters or kernels to calculate the features such as lines, edges and corners of an image. Normally, these filters are the mask matrix of numbers and moving over the input image to calculate the specific features. The convolution operation

of filters is also dot product and summation between the filters and input image. The output of these operations is usually passed through an activation function depending on its purpose, such as the Rectified Linear Unit (ReLU) activation function for non-linear input.

- **Pooling Layers** generally reduce the dimensional layers. These layers also represent the local or the global pooling layers. The pooling operation may be maximum, average and summation of each of a cluster of data at the prior layer.
- **Fully connected layers** are the final layers which connect every neuron in previous layer to every neuron on the next layer. The fully connected layer normally uses a softmax activation function in the output layer to classify the input image into several classes based on the training image.

III. Experimental Setup

A. Dataset

In this research, the data consists of two classes (N=8000): foot images (N=4000) and non-foot images (N=4000), as demonstrated in Fig. 6 and Fig. 7. In the case of foot images, these include different views, varieties of brightness, bare foot and wearing sock/shoe. In terms of non-foot images, these are randomized cropped images of a living room. Both classes of image are from online datasets Pascal VOC2012 [12] and image-net.org [13]. In the case of foot recognition using traditional methods, feature generation (LBPs and HOG) of images is required before processing in the training step to train and test the classifier model. In the training step of deep learning, the original data (100%) is divided into the training set (64%), the test set (20%) and the validation set (16%). The traditional classifier methods use 5-fold validation to evaluate the system.

B. Configuration

In the classification step, the training iteration for deep learning is set to 400, as configuration in Fig. 8. For other classifiers, K-nearest neighbors (kNN) algorithm classifies the test samples by using Euclidean distance with the number of k neighbor equal to k = 3, k = 5 and k = 7, respectively. The support vector machine (SVM) classifier predicts the class by using linear kernel.

C. Evaluation

Three measurement values are used to determine the performance of the foot recognition: sensitivity, specificity and complexity. The sensitivity measure is able to assess the accuracy of foot recognition as positive images. Specificity can evaluate the accuracy of non-foot recognition as negative images. Complexity can be used to evaluate the number of features, the number of connection nodes and Big-O-notation of each algorithm.

Fig. 6 Example of foot images

Fig. 7 Example of non-foot images

Convolutional Layer #1
Computes 16 features using a 3x3 filter with ReLU activation.
Padding is added to preserve width and height.
Input Tensor Shape: [batch_size, 200, 200, 3]
Output Tensor Shape: [batch_size, 200, 200, 16]

Pooling Layer #1
First max pooling layer with a 2x2 filter and stride of 2
Input Tensor Shape: [batch_size, 200, 200, 16]
Output Tensor Shape: [batch_size, 100, 100, 16]

Convolutional Layer #2
Computes 64 features using a 5x5 filter.
Padding is added to preserve width and height.
Input Tensor Shape: [batch_size, 100, 100, 16]
Output Tensor Shape: [batch_size, 100, 100, 16]

Pooling Layer #2
Second max pooling layer with a 2x2 filter and stride of 2
Input Tensor Shape: [batch_size, 100, 100, 16]
Output Tensor Shape: [batch_size, 50, 50, 16]

Dropout Layer
Add dropout operation; 0.6 probability that element will be kept

Logits layer
Input Tensor Shape: [batch_size, 128]
Output Tensor Shape: [batch_size, 2]

Fig. 8 Configuration on CNNs

IV. Results

A. Performance of Recognition

Table I shows the performance for foot recognition by using LBP, HOG with SVM or kNN classifier and compares it with CNNs methods. According to the result, HOG+kNN and CNNs provide higher sensitivity than LBP+kNN

TABLE I
SENSITIVITY OF FOOT RECOGNITION

Method	SENSITIVITY			
	SVM	kNN		
		k = 3	k = 5	k = 7
LBP				
r =1	0.74 ± 0.02	0.89 ± 0.01	0.88 ± 0.01	0.88 ± 0.01
r = 2	0.83 ± 0.03	0.90 ± 0.01	0.89 ± 0.02	0.89 ± 0.02
r =3	0.82 ± 0.02	0.89 ± 0.01	0.88 ± 0.01	0.87 ± 0.01
r =4	0.81 ± 0.02	0.88 ± 0.02	0.86 ± 0.02	0.86 ± 0.02
r=5	**0.86 ± 0.01**	0.88 ± 0.02	0.87 ± 0.01	0.87 ± 0.01
HOG	0.80 ± 0.12	**0.96 ± 0.01**	**0.97 ± 0.01**	**0.97 ± 0.01**
CNNs		**0.98 ± 0.01**		

Table II presents the performance for non-foot recognition (specificity). According to the results, both the kNN and the SVM with LBP features provide high specificity (equal or more than 0.92), while the combined with HOG feature performs lower specificity. CNNs have the lowest specificity in the experiment.

B. Complexity

Table III illustrates the number of each features where N_f is the number of features and N_c is number of convolutions in the neural networks. HOG provides the maximum number of features (N_f = 1764), while LBP with r = 1 has a lowest number of features (N_f = 10). CNNs has the highest complexity in the experiment due to its deep learning structure (N_c is approximately 88 million nodes.)

TABLE II
SPECIFICITY OF FOOT RECOGNITION

Method	SPECIFICITY			
	SVM	kNN		
		k = 3	k = 5	k = 7
C				
r =1	**0.96 ± 0.01**	0.92 ± 0.01	0.94 ± 0.01	**0.95 ± 0.01**
r = 2	0.83 ± 0.03	0.90 ± 0.01	0.89 ± 0.02	0.89 ± 0.02
r =3	0.92 ± 0.01	0.92 ± 0.01	0.94 ± 0.01	**0.95 ± 0.01**
r =4	0.92 ± 0.01	0.92 ± 0.01	0.94 ± 0.01	**0.95 ± 0.01**
r=5	0.92 ± 0.02	0.93 ± 0.01	**0.95 ± 0.01**	**0.95 ± 0.01**
HOG	0.59 ± 0.16	0.42 ± 0.02	0.41 ± 0.02	0.41 ± 0.02
CNNs		**0.03 ± 0.01**		

TABLE III
NUMBER OF FEATURES AND NUMBER OF CONVOLUTIONS

Features	Number of Features (N_f)	Number of Convolutions (N_c)
LBP		
r =1	10	-
r = 2	18	-
r =3	26	-
r =4	34	-
r=5	42	-
HOG	1,764	-
CNNs	-	88,473,600

Table IV presents the complexity of each classifier. N_s is the number of samples for training, N_f is the number of features, k is the number of the nearest neighbor and N_c is number of convolutions in the neural network. According to the table, SVM has the lowest complexity of classifier in the experiment, while CNNs might provide the highest complexity of classifier depending on the number of N_c.

TABLE IV
COMPLEXITY

	Complexity
SVM	$O(N_f)$
kNN	$O(N_s * N_f + N_s * k)$
CNNs	$O(N_c)$

V. DISCUSSION AND CONCLUSION

CNNs and HOG+kNN provide higher sensitivity for foot images while their specificity is lower than LBP+SVM. This might be because the CNNs model in the experiment can detect foreground image but may not be suitable to detect the background image with variety of shapes, color, and texture of image. On the other hand, CNNs has the highest complexity of classifiers which depends on the number of convolutions in this test. The kNN algorithm also has high complexity of calculation as can be seen from Table IV because it has to calculate the Euclidean distance between each test sample with all the training dataset.

VI. FUTURE WORK

For future work, the foot recognition with other deep learning algorithms such as Generative Adversarial Networks will be studied and developed to get higher accuracy with lower complexity of the algorithm.

ACKNOWLEDGMENT

This project is supported by the British Council Newton Fund: Researcher Links Travel Grant 2015/2016.

REFERENCES

[1] S. Anwer, A. Alghadir. "Effect of isometric quadriceps exercise on muscle strength, pain, and function in patients with knee osteoarthritis: a randomized controlled study". J Phys Ther Sci. 2014;26(5):745-8.
[2] Vincent KR, Vincent HK. Resistance exercise for knee osteoarthritis. PM R. 2012;4(5 Suppl):S45-52.
[3] E. Ceseracciu, Z. Sawacha, C. Cobelli, "Comparison of markerless and marker-based motion capture technologies through simultaneous data collection during gait: proof of concept". PLoS One. 2014;9(3):e87640. Published 2014 Mar 4. doi:10.1371/journal.pone.0087640
[4] J. Kwak, S. Xu and B. Wood, "Efficient data mining for local binary pattern in texture image analysis", Expert Systems with Applications, vol. 42, no. 9, pp. 4529-4539, 2015.
[5] Ma, Y., Chen, X., Chen, G., Pedestrian detection and tracking using HOG and oriented-LBP features, Network and Parallel Computing, 2011, pp. 176-184.
[6] L. Hou, W. Wan, K. Han, R. Muhammad and M. Yang, "Human detection and tracking over camera networks: A review," 2016 International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, 2016, pp. 574-580. doi: 10.1109/ICALIP.2016.7846643
[7] Y. LeCun, Y. Bengio, and G. Hinton. (2015). "Deep learning. Nature", 521(7553), 436-444. https://doi.org/10.1038/nature14539
[8] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, 2002.
[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human recognition," Proc. IEEE Conf. Computer. Vision. Pattern Recognition.,

pp. 1-8, Jun. 2005.
[10] C. Cortes and V. Vapnik, "Support-Vector Networks,". Machine. Learning. Vol. 20, No.3 (September 1995), pp. 273-297, 1995
[11] L. Hu, M. Huang, S. Ke and C. Tsai, "The distance function effect on k-nearest neighbor classification for medical datasets", SpringerPlus, vol. 5, no. 1, 2016.
[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results", Host.robots.ox.ac.uk, 2012. (Online). Available: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/. (Accessed: 10- OCT- 2018).
[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge", International Journal of Computer Vision, vol. 115, no. 3, pp. 211-252, 2015.