Hand Joint Detection

Genius Machado, Hetal Mistry, Sharanya Dave
Seidenberg School of CSIS, Pace University, New York City, New York
gm79361n@pace.edu , hm79869p@pace.edu , sd98623n@pace.edu

Abstract*—Osteoarthritis (OA) describes a degenerative joint disorder that is prevalent among older people and typically results in swollen and inflamed joints. The aim of this paper is to develop a method using images, videos and thermal data of 100 patients taken at Keio University Hospital to detect OA in hands. By using hand pose estimation on the video data, joint angles can be calculated and subsequently transformed into feature vectors. For the thermal and RGB images, hand key point detectors were trained to identify and crop the appropriate joints within the images. The resulting extracted features are combined and further trained on Support Vector Machines and Convolutional Neural Networks to obtain the final binary classification for each joint. While the proposed method generally shows favorable accuracy and F1-scores on the Proximal (PIP) and Distal Interphalangeal (DIP) joints, the performance on the Metacarpophalangeal (MCP) joints is limited by the low occurrence of affected joints in the dataset. We further compare the different modalities and found that, apart from the combined approach, using video data provides the best results. Clinical Relevance— The proposed method shows promising first results for the usage of visual and thermal data in combination with machine learning in order to detect OA.

Keywords: Hand Detection, Computer Vision, Machine Learning, Deep Learning, You Only Look Once (YOLO), Convolutional Neural Networks (CNN).

I. Introduction

Osteoarthritis (OA) is the most common form of arthritis, affecting millions of people worldwide. It occurs when the protective cartilage that cushions the ends of the bones wears down over time. Research in osteoarthritis (OA) pathology has been necessary for years due to its high economic impact, disability, pain, and severe impact on the patient's lifestyle. Although osteoarthritis can damage any joint, the disorder most commonly affects joints in your hands, knees, hips and spine. [5] OA goes beyond anatomical and physiological alterations (joint degeneration with gradual loss of joint cartilage, bone hypertrophy, changes in the synovial membrane, and loss of joint function) since cellular stress and degradation of the extracellular cartilage matrix begin with micro-and macro-injuries. Generally, OA is associated with aging. However, there are other risk factors namely obesity, lack of exercise, genetic predisposition, bone density, occupational injury, trauma, and gender. OA affects nearly 240 million people worldwide. According to the World Health Organization (WHO), by 2050, approximately 20% of the world's population will be over 60 years old. Of that percentage, 15% will have symptomatic OA, and one-third of these people will be severely disabled.

The first step to identify hand OA is to identify hand joints. Twelve joints are labeled by centered bounding boxes, each with a particular angle that aligns with the finger direction. The 12 joints include the Metacarpophalangeal (MCP), Proximal Interphalangeal (PIP), and Distal Interphalangeal (DIP) joints on each finger, excluding the thumb as the information from the other four fingers is mainly used for hand OA diagnosis.

II. Methods

A. Convolutional Neural Network (CNN)

A convolutional neural network (CNN) is a deep learning method based on a multilayer structure of artificial neural networks. A typical CNN architecture for image processing consists of stacks of multiple convolution filter layers and a series of data reduction or accumulation layers. Convolution layers are used as small-scale detectors to explore the features of an image, and a feature in a given location within the image can be computed by convolution. CNN applications offer a promising approach and are particularly successful when applied to feature extraction and classification. A great deal of data is required for the implementation of a CNN method, and the quality of the images, the availability of sufficient data, and an appropriate design of the CNN are important factors in creating a successful model. CNN expands the size of the layers alongside the initiating highlight. Mean pooling is utilized for reducing, by two stages, the image created from the convolutional layers. Further, the image is smoothed and taken care of by a method known as Multilayer Perception (MLP). [2] Then again, CNN snatches advantage of the neighborhood qualities of the images; for example, it manages input pixels which are situated close as well as far.

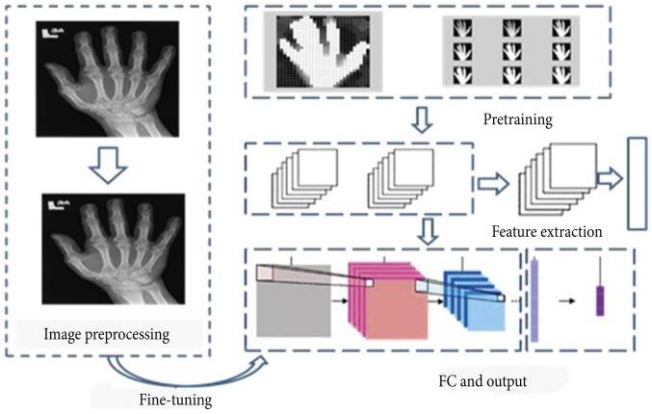

Fig. 1. This is a figure caption. It appears directly underneath the figure.

Convolution Layer: It is utilized to makes new images which is called highlight maps. The capacity map stresses the surprising parts of the primary image. As opposed to the information layer, the convolution layer is fairly extraordinary by the way it measures input which does not use connection loads and a weighted sum; then again, this layer utilizes filters that change images. These are called spatial channels.

Pooling Layer: The measurements of the image are reduced by the layer of pooling as it presents a combination of adjoining pixels into one delegate worth of a specific

image area. The technique of pooling is a standard technique as compared to other diverse image preparing frameworks.

Fully Connected Layers: They are considered once the Pooling and Convolution layers are added. In a completely connected layer, we connect neurons to all the arrangement of the past layer.

B. You Only Look Once (YOLO)

YOLO is a convolutional neural network algorithm, which is highly efficient and works tremendously well for real-time object detection [1]. A neural network not only helps in feature extraction, but it can also help us understand the meaning of gesture and help to detect an object of interest. A similar approach was adopted in [1], but the main difference is that in [3] it was done through Light YOLO, which is completely different from YOLOv3. Light YOLO is preferred for applications built on RaspberryPi. It achieves a good frame per second (fps) rate as compared to YOLOv3. However, the accuracy of YOLOv3 is 30% better when compared to Light YOLO, and as the application is not focused on the Internet-of-Things (IoT) product, YOLOv3 is preferred.

The training process first requires collecting the dataset, and after that the next step is to label it, so we use YOLO annotation to label our data, which gives us some values that are later explained in the model process. After that, when the data is labeled, we then feed it to the DarkNet-53 model, which is trained according to our defined configuration. The image is captured through the camera that can be an integrated (primary) camera or it can be any external (secondary) camera. Other than that, the application can also detect gestures from a video input as well [3].

After capturing real-time objects with the help of the OpenCV module, which is an open-source computer vision library, we can capture images and after that, we send them frame by frame from the real-time objects. Because of incorrect filtering, our dataset of collected images currently has a variable number of images. These images are labeled according to the classes we have stored for our YOLO algorithm, so we have successfully attained the coordinate and the class for our image set. After that, we can now set towards the training section. We then pass this set to our training algorithm, which is a deep neural network model YOLO.

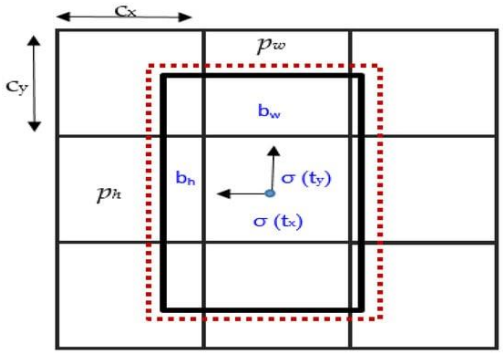

Fig. 2. The bounding box highlighting positions of the values for further classification.

Further, we discuss the methodology how YOLO deals with the network desired output which is achieved by using a formula which takes different co-ordinates, such as pw, ph, tx, ty, tw, th, cx, and cy as shown in Figure 2. These are the variables which we use for the bounding box dimensions. Obtaining the values of the boundary box (x-axis, y-axis, height, and width) is described by Equation (1).

$$\begin{aligned} b_x &= \alpha\ (\mathrm{t}_x) + \mathrm{Cx} \\ b_y &= \alpha\ (\mathrm{t}_y) + \mathrm{Cy} \\ b_w &= p_w e^t w \\ b_h &= phe^t h \end{aligned} \tag{1}$$

where bx, by, bw, and bh are the box prediction components, x and y refer to the center co-ordinates, and w and h refer to the height and width of the bounding box. Equation (1) used in the YOLOv3 algorithm shows how it extracts the values from the image in the bounding box, and below is the diagram of how these values are extracted from the bounding box.

The mechanism for the Yolo algorithm employs the use of a single neural network that takes a photograph as an Input and attempts to predict bounding boxes and class labels for each bounding box directly. To simplify the experiment design, if an image is input to the model, it first goes through a library before proposing the final three layers of convolutional data. The image is then passed to the YOLO network, which then further identifies the required target. After doing that it sends it forward to the feature map prediction block, where it further extracts the information which is required to identify the gesture, then, after predicting it, sends it to the decoding part, where the output predicted is mapped onto the image and then displayed.

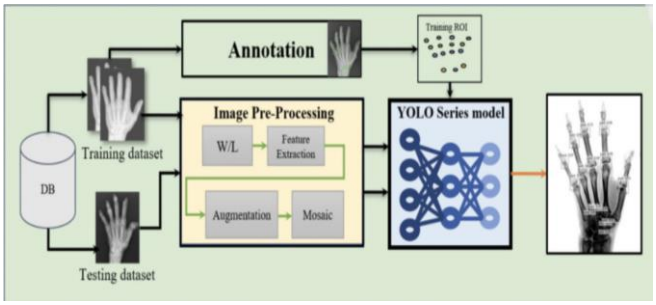

Fig. 3. The image processing pipeline for OA using YOLO series models

The feature map of YOLO is directly output as shown in Figure 3. Different channels are usually employed to represent different features of an image. The different channels of YOLO can not only express image features, such as joints in this study, but also express coordinates and confidence.

III. RESULTS

In the imaging examination protocol, the two-hand X-ray image is the most common imaging protocol for saving imaging time and reducing the X-ray exposure for patients. However, using the two-hand X-ray imaging protocol could result in shadows due to overlaying by other bones occurring in joint spaces as a function of the large angle of projection from the X-ray emission center. When using a one-hand X-ray, the shadows in joint spaces are minimized because the projection angle is smaller. According to our verification, the accuracy of the one-hand X-ray image was higher than that of the two-hand X-ray image.

The feasibility and effectiveness of the proposed YOLO-based model were verified to detect and locate the area of joint space narrowing in X-ray hand images.

Fig. 4. YOLO: The hand joint detection best result of the image contained right hand. It could fully detect 15 ROIs (Region Of Interest) in the X-ray image.

Fig. 5. CNN: The hand joint detection best result of the image contained right hand. It could fully detect 14 ROIs (Region Of Interest) in the X-ray image.

Models	Images	Learning Rate	Precision (%)	Accuracy (%)
YOLO	332	0.001	90.42	83.68
CNN	332	0.1	73.95	79.00

Table 1. Comparison Of The Accuracy Results For The Models Used For Hand Joint Recognition

IV. Author Contributions

Conceptualization: Genius Machado, Hetal Mistry, Sharanya Dave; Methodology: Genius Machado, Hetal Mistry, Sharanya Dave; Software: Hetal Mistry, Sharanya Dave; Validation: Hetal Mistry, Sharanya Dave; Formal analysis: Hetal Mistry, Sharanya Dave; Writing original draft preparation: Hetal Mistry, Sharanya Dave.; Writing review and editing: Hetal Mistry, Sharanya Dave.

V. Conclusion

CNN performs detection on various region proposals and end up performing predictions multiple times of various regions of an image, the image passes through the CNN once and then the output gives the prediction. On the other hand, YOLO architecture is more like a fully connected convolutional neural network. YOLO makes less than half the number of background errors as compared to Deep Learning CNN. YOLO architecture enables end-to-end training and real-time speed while maintaining high average precision. CNN offers end-to-end training as well but involves much more steps as compared to YOLO. CNN focuses on speeding up the framework by sharing computation and using neural networks to propose regions instead of Selective Search. While YOLO offers promising speed and accuracy over CNN.

VI. References

1. Wang HJ, Su CP, Lai CC, Chen WR, Chen C, Ho LY, Chu WC, Lien CY. Deep Learning-Based Computer-Aided Diagnosis of Rheumatoid Arthritis with Hand X-ray Images Conforming to Modified Total Sharp/van der Heijde Score. Biomedicines. 2022 Jun 8;10(6):1355. doi: 10.3390/biomedicines10061355. PMID: 35740376; PMCID: PMC9220074.
2. G. S. Mate, A. K. Kureshi, and B. K. Singh, “An efficient CNN for hand x-ray classification of rheumatoid arthritis,” Journal of Healthcare Engineering, vol. 2021, Article ID 6712785, 10 pages, 2021.
3. Mujahid, A.; Awan, M.J.; Yasin, A.; Mohammed, M.A.; Damaševičius, R.; Maskeliūnas, R.; Abdulkareem, K.H. Real-Time Hand Gesture Recognition Based on Deep Learning YOLOv3 Model. Appl. Sci. 2021, 11, 4164. https://doi.org/10.3390/app11094164
4. Oudah, M.; Al-Naji, A.; Chahl, J. Hand Gesture Recognition Based on Computer Vision: A Review of Techniques. J. Imaging 2020, 6, 73.
5. J. J. Andrade Guerreiro, Y. Aoki, S. Saito and K. Suzuki, "Detection of Osteoarthritis from Multimodal Hand Data," 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2022, pp. 3607-3610, doi: 10.1109/EMBC48229.2022.9871560.
6. M. Backhaus, T. Kamradt, D. Sandrock et al., “Arthritis of the finger joints: a comprehensive approach comparing conventional radiography, scintigraphy, ultrasound, and contrast-enhanced magnetic resonance imaging,” Arthritis & Rheumatism, vol. 42, no. 6, pp. 1232–1245, 1999.
7. Kulikajevas, A.; Maskeliunas, R.; Damaševičius, R. Detection of sitting posture using hierarchical image composition and deep learning. PeerJ Comput. Sci. 2021, 7, e442.
8. Yu, H.; Fan, X.; Zhao, L.; Guo, X. A novel hand gesture recognition method based on 2-channel sEMG. Technol. Health Care 2018, 26, 205–214.
9. Girshick R., Donahue J., Darrell T., Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Columbus, OH, USA. 23–28 June 2014; pp. 580–587.
10. Sismananda, P.; Abdurohman, M.; Putrada, A.G. Performance Comparison of Yolo-Lite and YoloV3 Using Raspberry Pi and MotionEyeOS. In Proceedings of the 8th International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, 4–5 August 2020; pp. 1–7.
11. Połap, D. Human-machine interaction in intelligent technologies using the augmented reality. Inf. Technol. Control 2018, 47, 691–703
12. Yan, S.; Xia, Y.; Smith, J.S.; Lu, W.; Zhang, B. Multiscale Convolutional Neural Networks for Hand Detection. Appl. Comput. Intell. Soft Comput. 2017, 2017, 9830641.
13. Ashiquzzaman, A.; Lee, H.; Kim, K.; Kim, H.-Y.; Park, J.; Kim, J. Compact Spatial Pyramid Pooling Deep Convolutional Neural Network Based Hand Gestures Decoder. Appl. Sci. 2020, 10, 7898.
14. Tran, D.; Ho, N.; Yang, H.; Baek, E.; Kim, S.; Lee, G. Real-time hand gesture spotting and recognition using RGB-D camera and 3D convolutional neural network. Appl. Sci. 2020, 10, 722.