**Recognition of hand sign images acquired with built-in camera of cell-phones**

Prepared by Asif Irfanullah Masum under the guidance of Ayatullah Faruk Mollah

*Department of Computer Science and Engineering, Aliah University*

*IIA/27 New Town, Kolkata 700160, India*

**Chapter One: Introduction**

Communication has been a vital point of human civilization and technological advancement. Hand Gesture recognition is a form of communication through which a machine can understand and extract information through hand sign images. It will facilitates a disabled person to communicate with a device without convention input devices. Hand gesture can be defined as the movement of hands and fingers in a particular orientation to convey some meaningful information like pointing to some object through index fingers, expressing victory sign or OK sign, waving hands, grasping an object, etc. Symbolic hand gestures represent some specific symbols like ’OK’ sign or gesture that represents numeric symbol ’1’ (raising the index finger and bending all other fingers). In most of the cases, these gestural movements conveys single meaning in each culture having very specific and prescribed interpretations. These gestures are observed in the spatial domain and are called static hand gestures characterized by the position of fingers (finger joint angle, orientation, and finger bending information). Unlike static hand gestures, dynamic gestures are considered in the temporal domain, presenting gesture as a sequence of hand shapes which includes starting through ending hand pose (e.g., hand waving, boxing).

The recent advancements in stereo vision camera that utilizes depth perception from smaller to larger distances have opened a huge scope for the researchers to work with depth information. Traditional web cameras do not provide the depth values (the distance of the gesturing hand from the camera). Depth information can help eliminating occlusion problems easily and can quicken the segmentation process with less error. In an occluded background, using depth information it is possible to extract the gesturing hand movement information including other important features (e.g., finger bending information) which can be effectively utilized in feature representations. Moreover, static gesture can be performed by the users with varying hand size, changes in hand position (orientation, rotation), and different illumination conditions. Scale-Invariant Feature Transform (SIFT) is an algorithm that works better for these types of variation. The algorithm generates key points from images and provides 128-dimensional feature vectors.

There has been novel works on hand gesture recognition and with the development of technology it is becoming easily accessible by people. There have been a lot of work on this field using depth image analysis which have yield accurate results. Our primary motive for this work is to create a model that can work on images which are readily available. We are focusing on RGB images which are clicked either on a inbuilt webcam or built-in camera of mobile phones. Hence making it more practical and inexpensive for people.

**Chapter Two: Related Works and Motivation**

Human hand is a highly articulated model, prominent in making deft poses. To recognize those hand poses many research works have utilized RGB cameras and applied either template-based approaches or model-based approaches on RGB images. Conventional RGB image-based gesture recognition techniques need to consider many research challenges, such as light sensitivity, cluttered background, and occlusions. However, the recent emergence of depth sensors has given an opportunity for the researchers to utilize the depth information in order to overcome those challenges. The depth data stream provided by the depth sensors (e.g., Microsoft Kinect, Intel Real Sense, Asus Xtion Pro) corresponding to the hand gesture images has given new dimensions to conduct research in hand segmentation process, finger identification techniques, finger joint detection, and finger tracking. Depth value indicates the distance of the gesturing hand from the RGB-Depth (RGB-D) camera in millimeters appropriate to make the segmentation process faster. Among the depth sensors, we have used Microsoft Kinect depth sensor that captures depth image in  resolution in a frame rate of 30 fps and 11-bit depth under the environment consisting of any ambient light. Depth information helps to extract additional features which can significantly improve the recognition results. Many researchers have developed depth sensor-based applications like interactive displays through Kinect, a system for therapeutic interventions, robot navigation through gestures, Kinect-based American Sign Language (ASL) recognition, etc. Other different applications of Kinect depth sensor includes categorizations of indoor environments by mobile robots equipped with Kinect , measuring canopy structure for vegetation, just to name a few.

From the depth sensors, the most common features used in hand posture recognitions are skeleton joint positions, hand geometry, hand-finger shape, area, distance features, depth pixel values, etc. Generally, these features can be categorized as local features or global features. The major challenges of these feature descriptors are variations of gesturing hands while articulating an emblem or symbolic gesture. A gesture may slightly differ in terms of hand shape and size, variations in translation, or rotation of the fingers for the same gesture. A robust hand gesture recognition system should be invariant to the scale, speed, and the orientation of the gesture performed.

The approaches that are followed by static gesture recognition system from binary images and time-series curves of those do not facilitate the possibility of extracting local finger context information. The authors have captured RGB images from webcam and converted them to binary images and applied SIFT algorithm to determine the recognition accuracy. In binary images, the finger context information, shape, orientation, bending fingers, and occlusion, cannot be preserved, a limitation that can be overcome by utilizing depth map information of the gesturing hand. SIFT keypoints are important feature points which are well distributed and contain information about not only thumb and baby fingers but also finger bending information of index, middle, and ring fingers. SIFT works on local oriented features rather than topological shapes of opening fingers which are considered as the global features. The edit-distances are calculated to apply distance-based matching algorithm, such as Finger-Earth Mover’s Distance (FEMD). Edit-distance-based matching algorithms are not completely rotation, orientation invariant because they are measured by comparing time-series trajectories based on the proximity distance and not based on the local shape information.

**Chapter Three: Development of Dataset**

The Dataset we are using is prepared using RGB images of hand sign which are taken with built-in camera of cell-phone (Model - iPhone 7) with resolution x720p . There are total of 1500 images constituting 150 RGB images of every single gesture. Figure 1 shows the gesture and corresponding images.

15 person x 10 samples for 10 classes

|  |  |  |  |
| --- | --- | --- | --- |
| IMG_0004 | IMG_0011 | IMG_0010 | IMG_0006 |
| IMG_0017 | IMG_0015 | IMG_0025 | IMG_0016 |

Fig. X. Data acquisition modality

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IMG_1696 | IMG_1770 | IMG_1752 | IMG_1743 | IMG_1733 |
| IMG_1706 | IMG_2076 | IMG_1806 | IMG_1816 | IMG_1835 |

**Fig 1.** Sample Images of dataset comprising of all gestures

Dataset is comprised of RGB images of gestures 0 to 9 with a uniform background with minumum background noise and collected with simililar light exposure of 15 people contributing 100 images each with 10 images for each gesture with some difference in the orientation.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| IMG_2140 | IMG_1846 | IMG_1988 | IMG_2313 | IMG_1679 | IMG_2474 | IMG_2130 | IMG_1394 |
| Gesture(0) | | | | | | | |
| IMG_1327 | IMG_1477 | IMG_1604 | IMG_2064 | IMG_1913 | IMG_2390 | IMG_2225 | IMG_2554 |
| Gesture(1) | | | | | | | |
| IMG_1462 | IMG_1311 | IMG_2539 | IMG_1590 | IMG_2044 | IMG_1901 | IMG_2372 | IMG_1756 |
| Gesture(2) | | | | | | | |
| IMG_1447 | IMG_1577 | IMG_2032 | IMG_1736 | IMG_2196 | IMG_1886 | IMG_2532 | IMG_2356 |
| Gesture(3) | | | | | | | |
| IMG_1721 | IMG_2018 | IMG_2339 | IMG_1433 | IMG_1868 | IMG_1560 | IMG_2514 | IMG_2193 |
| Gesture(4) | | | | | | | |
| IMG_1708 | IMG_1551 | IMG_2009 | IMG_1425 | IMG_1854 | IMG_2156 | IMG_2329 | IMG_2170 |
| Gesture(5) | | | | | | | |
| IMG_2079 | IMG_1335 | IMG_1927 | IMG_1491 | IMG_1621 | IMG_1784 | IMG_2402 | IMG_2253 |
| Gesture(6) | | | | | | | |
| IMG_1794 | IMG_1936 | IMG_1502 | IMG_1631 | IMG_2282 | IMG_2422 | IMG_2586 | IMG_2089 |
| Gesture(7) | | | | | | | |
| IMG_1808 | IMG_1646 | IMG_1516 | IMG_1949 | IMG_2107 | IMG_2433 | IMG_2596 | IMG_2283 |
| Gesture(8) | | | | | | | |
| IMG_1827 | IMG_1532 | IMG_1661 | IMG_2452 | IMG_1976 | IMG_2612 | IMG_2123 | IMG_2306 |
| Gesture(9) | | | | | | | |

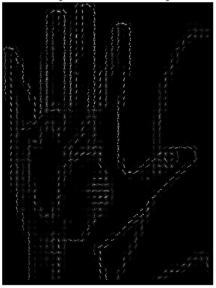
**Fig. 2** Variations in hand gestures

**Chapter One: Proposed Methodology**



**Fig 2.0 Dataflow of the model used for classification of hand gestures.**

The methodology we use to detect hand sign images consist of data pre-processing by the means of K-Means and then further segmentation of the image to extract the region of interest. This segmented images is used for feature extraction by the means of Histogram of Orient Gradients(HOG) features descriptors. We use these HOG feature descriptors in our classification process through Support Vector Machine classification model.

1. (b) (c) (d) (e)

**Fig. 3** HOG Feature Extraction

(a) Original Image (b) Gray Scale Image (c) K-Means pixel clustered image

(d) Segmented across the region of interest image (e) HOG Image.

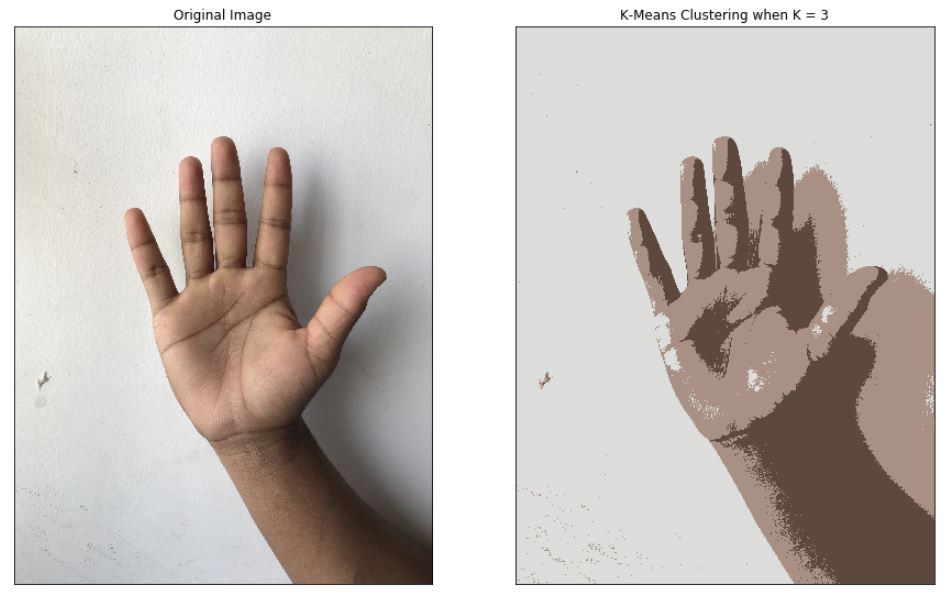
* 1. *K-Means Pixel Clustering*

For data pre-processing, we use K-Means Clustering to segment image into definitive color contours for further processing.

K-Means algorithm is an unsupervised algorithm to classify unknown data to some labels which may seem fit. It is used to cluster the datapoints of a given dataset into K-clusters or K-Centroids.

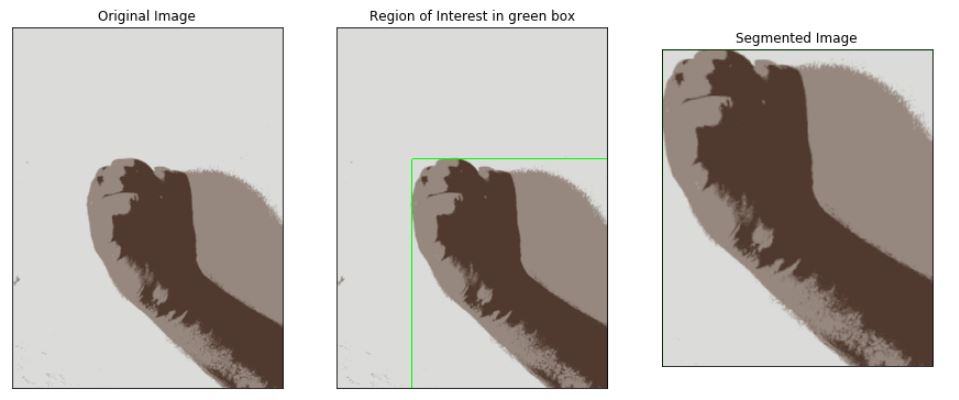
**Steps in K-Means algorithm:  
Step** 1: Number of clusters K are chosen intuitively.  
Step 2: Random centroids(not necessarily from given dataset) of number K are selected.  
Step 3: Closest data point to chosen centroids are chosen for that cluster.  
Step 4: New centroid are computed of each cluster that are formed.  
Step 5: Closest data point to new centroid are reassigned to form new cluster.

Step 6: If any reassignment took place in Step-5, re-iterate Step 4, otherwise the model is ready.



* 1. *Segmentation of ROI*

The clustered images are binarized using Otsu Thresholding along with normal binary thresholding technique. Then the connected contours of the binarized inages are located. Furthermore, using these contour we find the best fit rectangle for the maximum using bounding rectangle.



**Fig. X** Segmentation of the Region of Interest(ROI)

* 1. *Image dimension normalization*

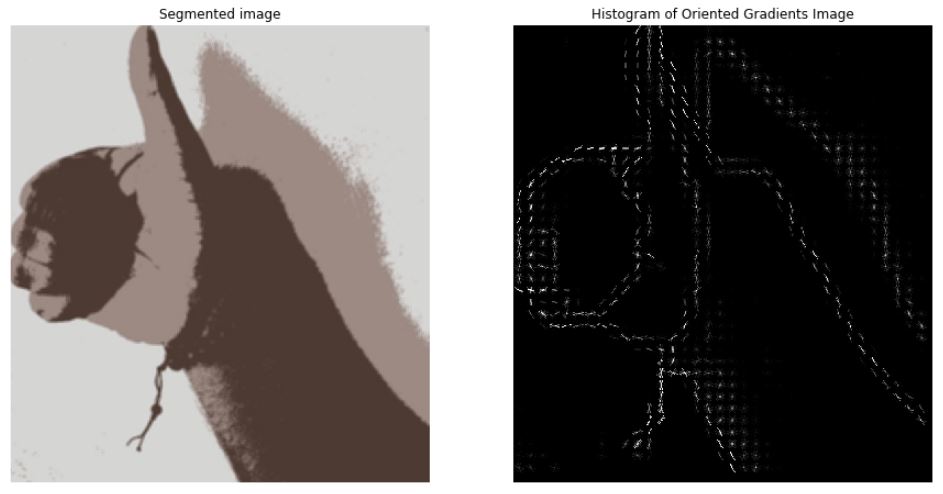
Image is re-sized to (X,X) grayscale images on which the feature extraction process is implemented

* 1. *Histogram of Oriented Gradients*

Histogram of oriented gradients (HOG) is a feature descriptor used to detect objects in computer vision and image processing. The HOG descriptor technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI).

Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.
2. Discretize each cell into angular bins according to the gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.
4. Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
5. Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.



**Chapter Five: Experimental Results**

**Table 1**. Performance of hand gesture classification for various classifiers and various cross-validation

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Cross Validation** | **Classifier** | **Accuracy** | **Precision** | **Recall** | **F-score** | **N-RMSE** | **AUC** |
| 3 | Gaussian Naïve Bayes | 0.9220 | 0.9299 | 0.9220 | 0.9231 | -0.7818 | 0.9636 |
| KNN | 0.9700 | 0.9710 | 0.9700 | 0.9695 | -0.5241 | 0.9991 |
| MLP | 0.9867 | 0.9852 | 0.9753 | 0.9827 | -0.3033 | 0.9984 |
| SVM  (Linear) | 0.9973 | 0.9974 | 0.9973 | 0.9973 | -0.0682 | 0.9999 |
| SVM  (Poly) | 0.9940 | 0.9942 | 0.9940 | 0.9940 | -0.2400 | 0.9999 |
| SVM  (RBF) | 0.9893 | 0.9900 | 0.9893 | 0.9893 | -0.1456 | 0.9999 |
| Random Forest | 0.7133 | 0.7505 | 0.7133 | 0.7133 | -1.7504 | 0.9347 |
| 5 | Gaussian Naïve Bayes | 0.9320 | 0.9406 | 0.9320 | 0.9330 | -0.6883 | 0.9655 |
| KNN | 0.9793 | 0.9799 | 0.9793 | 0.9791 | -0.5165 | 0.9991 |
| MLP | 0.9907 | 0.9871 | 0.9900 | 0.9867 | -0.5187 | 0.9996 |
| SVM  (kernel1) | 0.9980 | 0.9981 | 0.9980 | 0.9980 | -0.0577 | 0.9999 |
| SVM  (kernel2) | 0.9967 | 0.9968 | 0.9967 | 0.9967 | -0.0515 | 0.9999 |
| SVM  (kernel3) | 0.9920 | 0.9926 | 0.9920 | 0.9919 | -0.1108 | 1.0000 |
| Random Forest | 0.7100 | 0.7385 | 0.7100 | 0.7074 | -1.8152 | 0.9319 |
| 10 | Gaussian Naïve Bayes | 0.9367 | 0.9478 | 0.9367 | 0.9375 | -0.6100 | 0.9701 |
| KNN | 0.9813 | 0.9824 | 0.9813 | 0.9812 | -0.3883 | 0.9992 |
| MLP | 0.9927 | 0.9914 | 0.9900 | 0.9913 | -0.2718 | 0.9998 |
| SVM  (Linear) | 0.9980 | 0.9981 | 0.9980 | 0.9980 | -0.0408 | 1.0000 |
| SVM  (Poly) | 0.9980 | 0.9981 | 0.9980 | 0.9980 | -0.0408 | 1.0000 |
| SVM  (RBF) | 0.9947 | 0.9951 | 0.9947 | 0.9946 | -0.0527 | 1.0000 |
| Random Forest | 0.7060 | 0.7346 | 0.7060 | 0.7021 | -1.9104 | 0.9358 |

A **Support Vector Machine** is a yet another supervised machine learning algorithm. It can be used for both regression and classification purposes. But SVMs are more commonly used in classification problems (This post will focus only on classification). Support Vector machine is also commonly known as “Large Margin Classifier”.

Support Vector machines have some special data points which we call “Support Vectors” and a separating hyperplane which is known as “Support Vector Machine”. So, essentially SVM is a frontier that best segregates the classes.  
Support Vectors are **the data points nearest to the hyperplane, the points of our data set which if removed, would alter the position of the dividing hyperplane**. As we can see that there can be many hyperplanes which can segregate the two classes, the hyperplane that we would choose is the one with the highest margin.

We are not always lucky to have a dataset which is lineraly separable by a hyperplane. Fortunately, SVM is capable of fitting non-inear boundaries using a simple and elegant method known as [kernel](https://en.wikipedia.org/wiki/Kernel_method) trick. In simple words, it projects the data into higher dimension where it can be separated by a hyperplane and then project back to lower dimensions.

Here, we can imagine an extra feature ‘z’ for each data point “(x,y)” where z2=x2+y2  
We have in-built kernels like [rbf](https://en.wikipedia.org/wiki/Radial_basis_function_kernel), [poly](https://en.wikipedia.org/wiki/Polynomial_kernel), etc. which projects the data into higher dimensions and save us the hard work.

**Overall Accuracy**: 0.98 [**Support Vector Machine(Kernel=Linear)**]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gestures | Precision | Recall | F1-Score | Support |
| Gesture(0) | 1.00 | 1.00 | 1.00 | 40 |
| Gesture(1) | 1.00 | 1.00 | 1.00 | 41 |
| Gesture(2) | 0.95 | 1.00 | 0.97 | 38 |
| Gesture(3) | 1.00 | 0.96 | 0.98 | 53 |
| Gesture(4) | 1.00 | 0.96 | 0.98 | 52 |
| Gesture(5) | 0.96 | 1.00 | 0.98 | 44 |
| Gesture(6) | 1.00 | 1.00 | 1.00 | 42 |
| Gesture(7) | 1.00 | 1.00 | 1.00 | 46 |
| Gesture(8) | 0.96 | 0.94 | 0.95 | 51 |
| Gesture(9) | 0.93 | 0.95 | 0.94 | 43 |

**Confusion Matrix** [**Support Vector Machine(Kernel=Linear)**]

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Gesture(0)** | **Gesture(1)** | **Gesture(2)** | **Gesture(3)** | **Gesture(4)** | **Gesture(5)** | **Gesture(6)** | **Gesture(7)** | **Gesture(8)** | **Gesture(9)** |
| **Gesture (0)** | 40 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Gesture(1)** | 0 | 41 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Gesture(2)** | 0 | 0 | 38 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Gesture(3)** | 0 | 0 | 2 | 51 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Gesture(4)** | 0 | 0 | 0 | 0 | 50 | 2 | 0 | 0 | 0 | 0 |
| **Gesture(5)** | 0 | 0 | 0 | 0 | 0 | 44 | 0 | 0 | 0 | 0 |
| **Gesture(6)** | 0 | 0 | 0 | 0 | 0 | 0 | 42 | 0 | 0 | 0 |
| **Gesture(7)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 46 | 0 | 0 |
| **Gesture(8)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 48 | 3 |
| **Gesture(9)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 41 |

**Conclusion**

This paper describes a symbolic hand gesture recognition system and presents an effective way of utilizing Histogram of Oriented Gradients information of segmented clustered images of RGB hand signs images.

In future, we will analyze gesture recognition accuracy in terms of variations in cluster size using the principal component analysis (PCA), adaptive gray-scale levels, combining local, and global features (containing contour information) using hierarchical classification techniques.

**\References**

1. Soumi Paul, Hayat Nasser, Mita Nasipuri, Phuc Ngo, Subhadip Basu, et al.. A Statistical-Topological Feature Combination for Recognition of Isolated Hand Gestures from Kinect Based Depth Images. IWCIA, Jun 2017, Plovdiv, Bulgaria. ff10.1007/978-3-319-59108-7ff. ffhal-01517817f
2. Ahmed Abdal Shafi Rasel, Mohammad Abu Yousuf, "An Efficient Framework for Hand Gesture Recognition based on Histogram of Oriented Gradients and Support Vector Machine", International Journal of Information Technology and Computer Science(IJITCS), Vol.11, No.12, pp.50-56, 2019. DOI: 10.5815/ijitcs.2019.
3. Cho, Mi-Young, and Young-Sook Jeong. "Human gesture recognition performance evaluation for service robots." 2017 19th International Conference on Advanced Communication Technology (ICACT). IEEE, 2017.
4. Rahman, Md Abdur, and M. Shamim Hossain. "A gesture-based smart home-oriented health monitoring service for people with physical impairments." International Conference on Smart Homes and Health Telematics. Springer, Cham, 2016.
5. Yang, Geng, et al. "A Novel Gesture Recognition System for Intelligent Interaction with a Nursing-Care Assistant Robot." Applied Sciences 8.12 (2018): 2349.
6. Azeem, A., et al. "Hexagonal scale invariant feature transform (H-SIFT) for facial feature extraction." Journal of applied research and technology 13.3 (2015): 402-408.
7. Kashif, Muhammad, et al. "Feature description with SIFT, SURF, BRIEF, BRISK, or FREAK? A general question answered for bone age assessment." Computers in biology and medicine 68 (2016): 67-75.
8. Dalal, N.; Triggs, B., "Histograms of oriented gradients for human detection," Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol.1, no., pp.886,893 vol. 1, 25-25 June 2005