**COMP4336/9336 Light Sensor App report**
**z5172461 Huixian Zhou**

1.gesture recognition

(1).modalities

Gesture recognition technology, from simple to complex to fine, can be roughly divided into three levels: two-dimensional hand recognition, two-dimensional gesture recognition, three-dimensional gesture recognition.

Two-dimensional hand recognition, also known as static two-dimensional gesture recognition, identifies the simplest type of gesture. Only a few static gestures can be identified, such as a fist or a five-finger. This technique only recognizes the "state" of the gesture, not the "continuous change" of the gesture. In the end, it is a pattern matching technique that analyzes images by computer vision algorithms and compares them with preset image patterns to understand the meaning of such gestures. Therefore, the two-dimensional hand recognition technology can only recognize the preset state, the scalability is poor, the control feeling is weak, and the user can only realize the most basic human-computer interaction function.

Two-dimensional gesture recognition, still without depth information, stays on the two-dimensional level. This technique is a bit more complicated than two-dimensional hand recognition, which not only recognizes the hand shape, but also recognizes some simple two-dimensional gestures, such as waving at the camera. Two-dimensional gesture recognition has dynamic features that track the movement of gestures and identify complex movements that combine gestures and hand movements. Although this technology is no different from the two-dimensional hand recognition in hardware requirements, it can get more abundant human-computer interaction content thanks to more advanced computer vision algorithms. In the use experience, it has also improved a grade, from pure state control to a relatively rich plane control.

Compared to two-dimensional gesture recognition, three-dimensional gesture recognition adds a Z-axis of information that identifies various hand shapes, gestures, and movements. This gesture recognition with certain depth information requires special hardware to implement. Common is done by sensors and optical cameras. At present, there are mainly three kinds of hardware implementation methods, and advanced computer vision software algorithms can realize three-dimensional gesture recognition.

1.Structure Light: The basic principle of this technology is to calculate the position and depth information of an object by laser refraction and algorithm, and then restore the entire three-dimensional space. However, due to the calculation of the position depending on the displacement of the refracted light, this technique cannot calculate accurate depth information and has strict requirements on the identified distance.2. Time of Flight: The principle of flying time is to load a light-emitting component, capture the time of flight of the calculated photon through the CMOS sensor, calculate the distance of the photon flight based on the photon flight time, and obtain the depth information of the object. Computationally speaking, the flying time is the simplest in 3D gesture recognition and does not require any computer vision calculations.3. Multi-camera: This technology uses two or more cameras to simultaneously capture images. By comparing the differences between the images obtained by these different cameras at the same time, an algorithm is used to calculate the depth information for multi-angle three-dimensional imaging. Multi-angle imaging is the lowest hardware requirement in 3D gesture recognition technology, but it is also the most difficult to implement. Multi-angle imaging does not require any additional special equipment and relies entirely on computer vision algorithms to match the same target in both images. Compared with the disadvantages of high cost and high power consumption, the two techniques of structured light or optical flying time can provide three-dimensional gesture recognition effect of “cheap and good quality”.

(2)algorithms

In gesture recognition, the result of this algorithm requires a binary object, and a gesture either conforms to a predetermined gesture or does not match. Each gesture recognition algorithm requires different parameters, such as time intervals and thresholds. This is especially true when identifying specific gestures based on the process. Developers need to constantly test and experiment to determine the appropriate parameter values for each algorithm. This in itself is a challenging and tedious job. However, the recognition of each gesture has its own special problems.

There are three main types of static gesture recognition technology based on monocular vision:

The first type is template matching technology, which is the simplest identification technology. It matches the feature parameters of the gesture to be recognized with the pre-stored template feature parameters, and performs the recognition task by measuring the similarity between the two. "Hausdorff distance in the use of gesture recognition" uses the Hausdorff distance template matching idea to achieve gesture recognition. The edge images of the gesture to be recognized and the template gesture are transformed into the Euclidean distance space, and their Hausdorff distance or modified Hausdorff distance is obtained. The distance value is used to represent the similarity between the gesture to be recognized and the template gesture. The recognition result takes a template gesture corresponding to the minimum distance value.

The second type is statistical analysis technology, which is a classification method based on probability and statistics theory that determines the classifier by statistical sample feature vector. This technique requires people to extract specific feature vectors from the raw data and classify these feature vectors instead of directly identifying the original data. In the "Computer Vision-based Gesture Recognition Research", although the Hausdorff distance algorithm is also used, but the template gesture is not proposed, but the fingertip and center of gravity features are extracted for each image, and then the distance and angle are calculated. The gestures are separately calculated from the distance and the angle, and the digital features of the distribution are obtained. The Bayesian decision of the minimum error rate yields the values of the distance and angle used to segment the different gestures. After the classifier is obtained, the collected gesture images are classified and identified. The algorithm combining the multi-scale model and the moment delineator in this paper also uses statistical analysis techniques.

The third category is neural network technology, which has self-organization and self-learning capabilities, has distributed characteristics, can effectively resist noise and deal with incomplete modes and has the ability to promote patterns. With this technique, a training (learning) phase of a neural network is required before recognition. Among them, the BP neural network is more commonly used. BP (Error Back Propagation Neural Network) is a neural network that can self-organize in the direction of satisfying a given input-output relationship. When the actual output on the output layer is inconsistent with a given input. The descending method is used to correct the old bonding strength between layers until the final input and output relationship is satisfied. The error propagation direction is opposite to the direction of signal propagation, which is called error back propagation neural network. The theory of BP neural network believes that as long as the relationship between input and output is continuously given, in the process of learning neural network, the internal structure of this relationship will be formed inside, and the speed of relationship formation will be achieved. Practical value, then there is no difficulty in the application of BP. "Visual-based gesture recognition and its application in human-computer interaction" uses BP (error back propagation) neural network method based on direction histogram.

It can be seen that if the algorithm based on geometric classification is simple, compared with the method of neural network, it shows reliability, which allows defining a feature set of different gesture categories, and estimating a locally optimal linear discriminator according to the gesture image. A large number of features extracted in the corresponding gesture categories are recognized, but the efficiency of learning is not high. As the sample size increases, the recognition rate of the algorithm is not obvious. However, BP neural network requires a certain learning stage. There may be serious shortcomings such as the large number of intermediate layer neurons, the long learning time, and the wide range of coupling coefficients. In the practical application of BP neural network, the network structure should be considered reasonably. For example, based on Sift static gesture recognition, the skin-based Sift feature is used, the PCA is reduced in dimension, the HOG feature is used to express the gesture, and the PCA dimension reduction is also performed. The two features are used as feature vectors to train the classifier to generate a static feature recognition library.

1. Downs-up algorithm

In my code, I create a timer which refreshes the charts each 1000ms. All the data together with the time detected by the light sensor when the environment light changed are put in the list <lightlist>. Each 1 second I track the data of the Last

5seconds in the list. When the 2 adjacent data goes up if the up mark (int)up not equal to 1 then mark it as 1 and set the (int)bottom equal to the less one. Else if the up equal to one If the result large than 0.15 multiply bottom than the (int)ups plus one Vers visa. The min(ups, downs) is the result of the count Down-up.

2. Gesture- control

There are 3 type of gestures can be detected by my app:

Gesture1: down -> up

Gesture2: down -> up-> down -> up

Gesture3: down -> up-> down -> up ->down -> up

I designed a music player app and each gesture can control one function.

Gesture1: Play and pause.

Gesture2: Turn the volume up

Gesture3: Turn the volume down

3. Application performance

Indoor:

performed 20 gestures, 20 gestures are detected in which 3 are correctly recognized

miss rate = (20-20)/20 = 0%

recognition accuracy = 17/20 = 85%.

Outdoor:

performed 20 gestures, 19 gestures are detected in which 2 are correctly recognized

miss rate = (20-20)/20 = 0%

recognition accuracy = 18/20 = 90%.

The wrong detection of the gesture happened on Gesture2 and Gesture3 because the gap between 2 adjacent down-up is too tiny that the light sensor can't detect the change. I try to change the sensorManager function parameter: SENSOR_DELAY_GAME,SENSOR_DELAY_FASTEST,SENSOR_DELAY_GAME, SENSOR_DELAY_NORMAL, SENSOR_DELAY_UI.Anyway, it doesn't help apparently.

4. Challenges

The chart API I use (Hellocharts) does not have the function to refresh chart automatically. To refresh the line chart I set a Timer to refresh it every second. In the timer, I get the recent 1-second data in <lightlist> if There is no change then set the point data (current-time, last light data) and put it into line. If there is data in last 1 second I get the data and use the function addfirst() to put it into linked-list Because the get data order (last one -> first one) is opposite from the draw line order(first one ->last one). Then put the points to the line and draw the line. Use API function currentwindow() and maxwindows( ) to move the line to the right for 1 second.

After putting update UI instruction into Timer the app crashes without any error message. I've no idea what happened cause that and there is no clue it was caused by set textview message in timer. I searched online to learn about the thread in android and I find the new thread handler to handle the UI update. In timer, I send the update information through the message to handler and let handler deal with that.