Octaculus - 8-way directional gesture detection using machine learning and light dependent resistors

DAN SCOTT, ADAM MCDANIEL, and LOGAN WRINKLE, University of Tennessee, USA

Light dependant resistors are a cheap substitute for detecting gestures to control devices. Lighter, cheaper and and faster than cameras or cloud-point sensors, a small array of light dependent resistors can detect 8-way directional gesture movement. Additionally, the direction of the movement was modeled and calculated using machine learning models.

MAIN COMPONENT: Data Collection
SECONDARY COMPONENT: Feature Selection

Additional Key Words and Phrases: machine learning, gesture , direction, human computer interface, HCI

ACM Reference Format:
Dan Scott, Adam McDaniel, and Logan Wrinkle. 2022. Octaculus - 8-way directional gesture detection using machine learning and light dependent resistors. 1, 1 (December 2022), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Traditional touchless gesture recognition uses various devices like cameras or infrared point projection sensors. Cameras introduce a possible privacy concern and increase energy usage and cost. Both cameras and point projection sensors are suspected of having higher energy cost. An alternate method of determining simple gesture recognition using a small number of LDRs to sense the direction of movement under normal light conditions using machine learning models was found to be a suitable replacement for 8-way linear direction detection (up,down,left, right, up right, down right, up left & up right). In the long term this technology could provide a smaller and cheaper method of gesture recognition which could be adapted to a multitude of user interfaces. We tested three different configurations of LDRs: three in a triangle shape, four in a diamond or square shape, and five in a pentagon shape. We found that three LDRs is the minimum number of LDRs to detect 8-way direction movements. Additionally, three machine learning classifiers were compared for classifying 8-way, linear direction of movement. Of the three classifiers, one stood out as working well and had a small binary size footprint, the support vector classifier.

Authors' address: Dan Scott, dscott57@vols.utk.edu; Adam McDaniel, amcdan23@vols.utk.edu; Logan Wrinkle, lwrinkle@vols.utk.edu, University of Tennessee, Knoxville, TN, USA.

XXXX-XXXX/2022/12-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 DATA

The data for the project wholly consists of the voltages of the light dependent resistors on the measuring device. Because the measuring device is unique to the project, there aren't any existing data-sets on the subject. As a result, data will need to be collected using the measuring device once it is created. The data itself is very friendly to machine learning: the fast, periodic measurements of voltage across the light dependent resistors is directly proportional to the light received by the component. So, when someone's hand moves over the photo-receptors, it becomes very straightforward to determine the direction of the movement based on the relationships of the values reported by each of the LDRs. When a hand begins to move in a direction over the device, the sensors which read a voltage drop *first* determine the origin of the movement. When the hand passes over the device entirely, the sensors which read an electrical resistance change *last* determine where the hand moved relative to the origin. Additionally, gestures such as movement away from or towards the device can be detected by a consistent increase or drop across all of the sensors at once.

2.1 Data Collection

After building the fist prototype board, the 5-LDR placed in a pentagon arrangement (Appendix figure 3), an initial 10 samples of each direction were collected to ensure the data pipeline generated consistent results. This data proved useful for exploratory analysis which uncovered general knowledge of the data points for each resistor in each configuration. Figure number 1 shows a heat map for the raw data by each LDR. It can be visually noticed that the data shows variance by LDR channel over time and each LDR peaks at different points in time. This is expected as each LDR is spatially separated by some distance and it takes some time to move an object or hand across the LDRs. The table below depicts a heat map of the raw values from the 3 LDR configuration of an 'up' direction. Other events showed similar heat maps, however the peaks were noticeably different for each direction and each LDR.

After the initial data collection (10 events X 8 directions), an additional 100 samples were generated for the 5-LDR configuration. These initial events were recorded at approximately 36 Hz. In this configuration a direction event took 16 frames of data, or about 1/3 of a second. This was typical of the speed of movement over the LDRs throughout all the samples produced. However, since this 36 Hz configuration did not produce a high test-accuracy model, therefore the configuration for recording the events was changed to 100 Hz samples.

Because we changed the frequency of sampling, all data had to be re-collected at the new frequencies. After changing to the 100 Hz recording speed, 3200 (400 X 8 directions) samples were recorded for the 3-LDR configuration. 800 (100 X 8) for the 4-LDR and 800 (100 X 8) for the 5-LDR configurations were recorded. We found this time consuming and tiring because we used manual

Time	LDR 0	LDR 1	LDR 2
1669577434.72	4928	5120	4928
1669577434.73	4928	5120	6400
1669577434.74	4928	5248	10432
1669577434.75	4928	9344	12352
1669577434.76	4928	11904	13376
1669577434.77	4992	13248	14016
1669577434.78	9280	14080	14464
1669577434.79	12224	14720	14848
1669577434.80	13888	15040	14976
1669577434.81	14848	15232	15040
1669577434.82	15424	15488	15168
1669577434.83	15936	15616	15168
1669577434.84	16256	15680	15040
1669577434.86	16448	15744	15104
1669577434.87	16704	15808	14976
1669577434.88	16768	15744	14976
1669577434.89	16896	15808	14848
1669577434.90	16960	15680	14656
1669577434.92	16960	15680	6400
1669577434.93	17024	6400	5184
1669577434.94	16960	5376	5056
1669577434.95	16960	5248	4992
1669577434.96	6528	5184	4992
1669577434.98	5248	5120	4928

Fig. 1. 3-LDR Heat Map "Up" direction

methods to create the samples. Because of the human nature to move during long periods of standing of repetitive movements, additional variance was knowingly included in the samples. As stated earlier, this variance, coupled with multiple light sources required yet a third, more controlled recording of data samples.

The final data collection was from a single point light source, with all other lights off and secondary light sources like windows and doors closed. Additionally, the single point light source was stationed directly orthogonal to the breadboard, about 3 feet overhead. The data collection in this environment was 1600 events (200 X 8 directions) for the 3-LDR, 1600 events (200 X 8 directions) for the 4-LDR and 800 events (100 X 8 directions) for the 5-LDR confuration.

3 METHODS

For each of the different configurations of light dependent resistors on the device (3-LDR, 4-LDR & 5-LDR), several features and machine learning models were tested and evaluated in like fashion. Each case was a classification problem, with 8 directions as the classification.

(1) First device consist of three LDRs in a triangle shape.
(2) Second device consist of four LDRs in a diamond shape.
(3) Final device consist of five LDRs in a pentagon shape.

3.1 Labeling Data

The data was collected by recording LDR values from the device while performing the desired hand gestures to be predicted. For each of the approaches above, the data was first labeled manually for each hand gesture performed during the recording. Careful collection of the data was necessary to restrict additional movement and subsequent sensor detection. This was found to be extremely difficult when multiple light sources were present, such as a room light, several windows and doors. To limit the recording of secondary movement (like movement from other body parts), A single point light source was used to record the samples. It was uncovered that even simple body movement, like shifting one's weight from one foot to another would change the LDR values due to multiple sources of light coming from windows or open doors.

During the exploratory analysis, it was determined the data could be labeled programmatically by detecting the percent of change over a time period. After the initial data was labeled, a programmatic method was developed to label the data. The programmatic method was to take the average over a specific period of time and when the average changed more than a specific percent, the 'event' would be labeled for that direction. Directional samples were collected separately for each direction and file naming conventions were used to identify both the direction as well as the configuration.

3.2 Features

The raw data from the LDRs was an approximate scale of the voltage change from 0 to 65235. This was due to the GPIO library and did not hinder the process of determining when an event occur ed. The scaled output is still linearly related to the voltage drop or rise due to light increase or decrease.

Early exploratory analysis indicated that events varied in length (time) per sample rate. Due to this variance, several window sizes were experimented with to determine both a usable detection of an event as well as suitable length of data to calculate individual features. In addition to the varying window size, the feature calculations for moving average, skew, kurtosis and eventually center-of-mass features were tested with each machine learning model. While several features were tried, eventually one stood out more than the others: center-of-mass.

The features for skew and kurtosis varied with window sizes from 10 to 40 frames (a frame is one sample) gave sub-par test-accuracy results and even worse validation results. The feature for center-of-mass varied with window sizes from 10 to 30 frames gave good results. Additional experiments revealed that either 18 or 20 frames of data for the calculations of the center-of-mass resulted in the highest test-accuracy.

The test accuracy for the random forest model was used as a proxy to determine the best window length. By using a grid-search with 10 fold cross validation for each of the windows lengths, the suitable window length to indicate the average change in conditions indicating an event as well as the parameters for the random-forest model were determined. Table 1 shows the iterative results in accuracy descending order to determine optimal event recognition (indicating an event) and optimal window length for calculating the center-of-mass feature. The number of samples used to indicate an event was best when the sample window was 20 items long. This seems to coincide with the actual length of a nominal event and appears intuitively to match the common length of an event. In layman's terms, because an event (pass of the hand over the LDRs) was typically no longer than 1/5th of a second, the frequency of the sampling of the LDR voltage change at 100 Hz would capture at least 1/5th of a second worth of data (20 frames).

Length of Event Window to Average	Center of Mass calculation Window	Model Accuracy
20	**20**	**96.14%**
12	20	96.11%
18	20	96.07%
15	20	96.00%
10	20	95.97%
13	20	95.93%
13	13	95.46%
15	30	92.33%
10	10	92.02%
5	10	91.61%
5	30	91.29%

Table 1. Search for best window sizes.

The detection of the event using the change of the average voltage over a window of 20 frames for each LDR is easy to calculate on the device platform (Raspberry Pi), conserves resources by not constantly needing to predicting a non-direction event and did not introduce significant latency of the sample program using the final models. Meaning, the non-direction class would not be needed. This allowed the models to be built without adding a 9th class to indicate no direction. It also made detecting the event consistent with the labeling method mentioned above. The algorithm was built to indicate an event when the any of the windowed averages change more than one percent (.01) for any of the LDRs. This also seems to eliviete the need to scale data with respect to ambient light levels. Some experimental data was produced to determine the effect of abrupt changes in ambient light levels and it was found to some effect on the average change calculation. Meaning, if you change the level of ambient light, like turning on a light or turning off a light, that change would be included in the average change calcuation and cause a pseudo event to be detected. The parameterization of this algorithm also makes the program to detect the event tune-able. However, we did not investigate the optimal tuning parameter for calculating what impact the percent change would have with regards to identifying events. It is assumed that lowering the percent change parameter would have some effect on the programs ability to detect events from non-events.

4 RESULTS

The capability to use low cost, and energy efficient LDRs makes integrating gesture control into consumer devices more controllable and affordable. The added benefit of no-touch controls emphasize hygienic options. Eliminating the privacy concerns of cameras to detect movement reduces consumer concerns. The proposed configuration should have a high degree of accuracy to allow consistent, dependable and accurate directional control.

Table 2 depicts the test accuracy for each of the configurations (3, 4 & 5 LDR) for three machine learning models using the center-of-mass feature. In all three configurations, the Support Vector Classifier scored a test-accuracy of 100%. This indicates the model performs superbly well regardless of the number of LDRs used. This seems to imply that at least three LDRs could be used, and that any additional number of LDRs would also work to detect and classify 8-way linear direction movement accross the LDRs - up, down,left, right and the four diagonal directions.

Model	3 - LDR	4 - LDR	5 - LDR
Random Forest	99.4%	96.3%	98.6%
Gradient Boost	98.8%	95.6%	98.1%
Support Vector	**100%**	**100%**	**100%**

Table 2. Accuracy of Machine Learning Models

4.1 Support Vector Classifier

Further investigation of the Support Vector Classifier (SVC) model uncovered that the number of support vectors decreases with an increase in LDRs. Also, the size of the model on disk, decreases with an increase in LDRs.

	3-LDR	4-LDR	5-LDR
Support Vectors	8	8	8
Number of vectors per Class	[20,10,7,8, 20,6,9,9]	[6,7,9,5, 5,7,10,5]	[7,5,4,6, 4,4,4,4]
Size of Model	9,750 bytes	7,239 bytes	5,131 bytes

Table 3. SVC Metrics by LDR configuration

5 TIMELINE

(1) Device Build and Program - Nov 8
(2) Data collection and cleaning - Nov 10
(3) Feature Engineering - Nov 13
(4) Machine Learning - Train, Test, Validate - Nov 18
(5) Draft Paper - Nov 19
(6) Final Paper - Nov 21
(7) Video Recording - Dec 1
(8) Presentation - Dec 2

6 LIMITATIONS

The device works very well with reliable input, but there are some limitations with the current setup of the photo resistors. If a person moves their hand over the detector with an arc in their movement, it will occasionally read the direction of the arc instead of the intended direction. This is an issue because the human arm tends to arc naturally. Additionally, because the device functions by detecting shadows, it is inherently limited by the environment it is used in. Shadows from people standing nearby could possibly affect the performance of the device depending on the lighting. Flickering lights may cause the device to perceive phantom gestures. Also, the current setup of the program reading into the model will sometimes read gestures twice when performed very slow, but to trigger this the user must deliberately perform the gesture across multiple seconds over the device. It may be possible to mitigate these phantom gestures by increasing the percent change needed to trigger a prediction, however this would make the entire model less sensitive to changes. Using a random forest classifier model with the data seems to make the device much more sensitive to the environment's lighting, but since switching to a support vector classifier model the device has been much more tolerant of various lighting setups. Now, the device can function in environments like a classroom using multiple lights, without depending on a specific point light source.

7 DISCUSSION

First, the data was collected in two environments by shining either a phone light or a high lumen light directly over the detector and performing gestures over the device. The device waits for a 0.1% change in the photo resistors' reported voltages to trigger a reading. The device records the sample and tags the recorded training data with the predetermined label. A validation data set was also recorded using the same method. When the sampling rate of the photo resistor voltages was increased, the data sets were rerecorded. A thousand samples were recorded for each configuration, along with a thousand additional samples for the triangular 3 photo resistor configuration. Despite the fact that the model was trained on using point light sources, albeit with ambient light still in the room, the model still performs well in environments without multiple light sources. This was particularly unexpected, due to the fact that multiple lights cast multiple shadows. The detection likely works well in multi-light environments because the extra shadows still move over the detector in the same direction, when the hand directly over and close to the sensors.

Originally, the second environment's training data was very different than the first environment's; this lead us to believe that the device would not function under other environments with different ambient lights. It was discovered, however, that there was a faulty light that was flickering in the room. When the light source was fixed, the training data immediately looked more similar to the first environment's.

The first features used to train a model from the 3, 4, and 5 photoresistor devices' data were skewness and kurtosis of the photoresistor voltages sampled at 36Hz over 400 millisecond windows (10 samples). This approach did not achieve good results when tested with our validation data set. After increasing the sampling rate from 36Hz to 100Hz, though, there was an immediate improvement in the results. Switching to center of mass instead of skewness and kurtosis made the biggest difference: the center of mass of the LDR voltages is not affected by intensity of the light source and so the model is less affected by the environment.

The 3 photo resistor configuration of the device seems to be the most reliable attempted implementation, which was very unexpected: the consensus was that the 5 photo resistor configuration would be much more accurate and reliable because it would have more information, or at least more redundancy, than the triangular photo resistor setup. It might also be due to the fact that the LDRs were placed somewhat close to each other on all of the device configurations (approximately one inch between neighboring photo resistors). At larger distances between neighboring photo resistors, it could be that the pentagonal LDR configuration outperforms the smaller configurations and the larger triangular configuration. It was thought that the difference in performance could be due to the additional samples in the training set, but even when trained with the same number of samples the 3 LDR configuration is best.

8 POSSIBLE APPLICATIONS

Directional movement is used in many human computer interfaces. Devices such as a mouse, a joystick controller a touch pad and even a keyboard use directional movement to send input to various devices. Using LDRs to provide direction input to devices is possible and has some advantages over those devices. They are static in place, do not require touch, are low cost and can be configured in several different ways. Directional input in some devices, such as those with touch screens could be augmented to included this directional input possibly eliminating the need for touch screens. While this was not part of this project, LDRs could also used as individual on/off buttons (binary input). Combining the use of the input as a binary switch in conjunction with directional gesture input opens even more possibilities.

9 FUTURE WORK

Our original hypothesis was that the 3-LDR setup would be much less accurate, however in practice it was nearly perfect in a controlled environment. However as stated in the Limitations section, it struggled with changes in ambient lighting, slow hand motions, and detecting the motion based off the tail end of the gesture (non-linear hand movements). Our future work will emphasize mitigating these limitations.

We will experiment with grids of LDR's and see how that effects the model's performance across variable use environments. Given that this grid method is what is most commonly used, we hypothesize that the grid would be more resistant to the types of issues we encountered. We will also experiment with variable distances between LDR's. Our initial goal was to see how few LDR's would be needed to generate accurate results and moving forward we would like to shrink the size of the system while maintaining high accuracy. We maintained a distance of approximately 1 inch between each LDR however in the future we will increase and decrease this amount and test its effectiveness. A larger spacing would likely give a larger window to detect movements, however this does not necessarily mean a higher accuracy. A smaller spacing may keep the accuracy in a controlled environment but be more susceptible to our existing lighting and movement issues. Finally we will try to retrain the model using samples from more natural hand gestures. We tried to control the environment as much as possible and as a result, we resisted the natural movement a hand would typically make.

Additionally, using a static window length to calculate the average change of the LDRs voltage seems to limit the speed of the movement detected. As mentioned above, slow hand movements accross the LDRs sent multiple event signals and thus multiple directions were indicated - like two ups, or two downs. It is possible to make the window length dynamic and capture the center-of-mass based on the new window size. This, however would require a different model, and perhaps multiple models for various lengths. We did not investigate this option, but noted this in our discussion.

10 CONCLUSION

Determining 8-way linear direction using light dependent resistors (LDRs) is possible with configurations of 3 or more LDRs. LDRs are suitable for detecting near proximity movement as objects intersect the light falling on the LDR. When LDRs are placed spatially near each other, the difference and change in resistance detected may be used to determine a linear direction. This paper concludes that 8-way linear direction (up, down, left, right, and the for diagonal directions between) can be ascertained using a machine learning model.

11 APPENDIX

This is the appendix of reference figures. The figures are for illustrating how the LDRs were arranged on the bread boards. Figure 2 depicts the pentagon arrangement of the 5 LDR on the bread board. Figure 3 depicts the diamond arrangement of the 4 LDR on the bread board. Figure 4 depicts the triangle arrangement of the 3 LDR on the bread board.

Finally, the general schematic for the 3 LDR is provided and how it was connected to the MPC3008 Analog/Digital controller. The MPC3008 uses a common SPI interface and the SPI interface to the raspberry pi is not shown in the diagram.

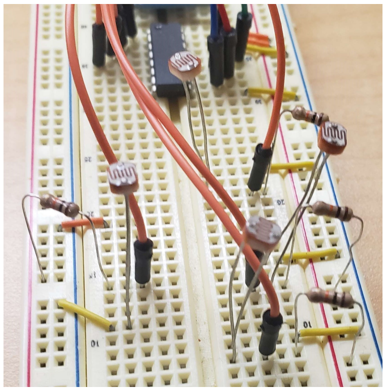

Fig. 3. 4 LDR Breadboard

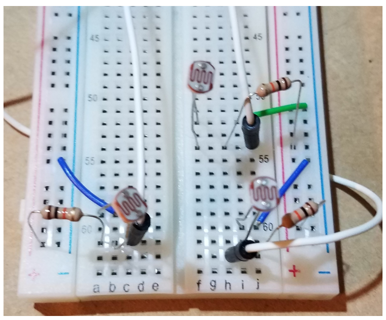

Fig. 4. 3 LDR Breadboard

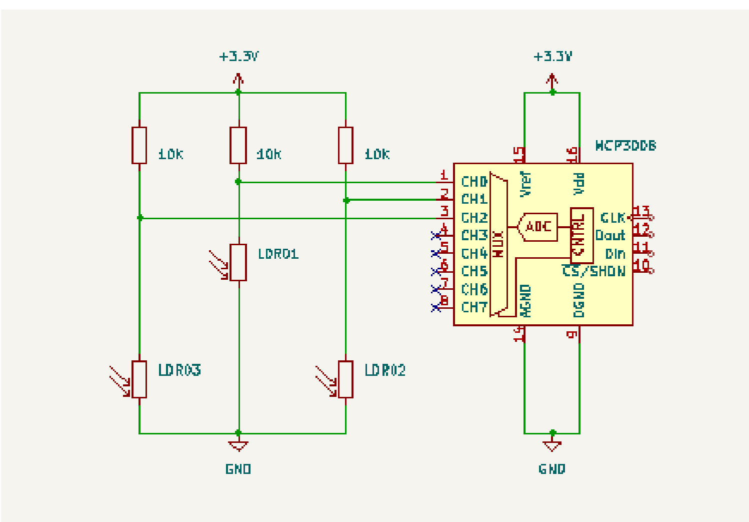

Fig. 5. 3 LDR Schematic

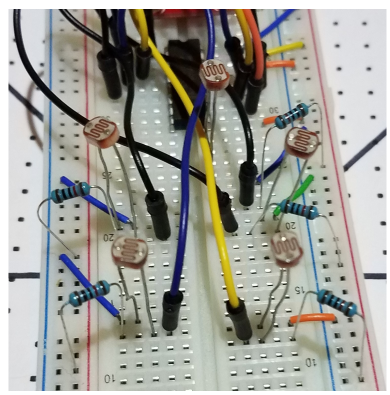

Fig. 2. 5 LDR Breadboard