Defining Stable Touch Area based on a Large-Screen Smart Device in 3D-Touch Interface

https://doi.org/10.3991/ijim.v13i02.10153

YounghoonSeo, Dongryeol Shin, Choonsung Nam(✉)
Sungkyunkwan University, Suwon, Republic of Korea
namgun99@gmail.com

Abstract—Touch interface technologies for mobile devices are essentially in use. The purpose of such touch interfaces is to run an application by touching a screen with a user's finger or to implement various functions on the device. When the user has an attempt to use the touch interface, users tend to grab the mobile device with one hand. Because of the existence of untouchable areas to which the user cannot reach with the user's fingers, it is possible to occur for a case where the user is not able to touch a specific area on the screen accurately. This results in some issues that the mobile device does not carry out the user's desired function and the execution time is delayed due to the wrong implementation. Therefore, there is a need to distinguish the area where the user can stably input the touch interface from the area where the users cannot and to overcome the problems of the unstable touch area. Furthermore, when the size of the screen increases, these issues will become more serious because of an increase in the unstable touch areas. Especially, an interface that receives position and force data like 3D-touch requires the stable area setting different from the conventional 2D-touch. In this paper, we search and analyze the stable touch areas on the large screen where the user can do 3D-touch inputs.

Keywords—Touch screen interface, 2D-touch, 3D-touch, Stable 3D-touch area, Mobile device

1 Introduction

There has been a huge shift in smart device market in recent years, with the most notable sign of the sales of phablets, a compound word of phone and tablet. IDC suggests that starting in 2018, the sales of phablets or smart devices with the screen size of between 5.5 inches and 7 inches will exceed those of other sizes [1]. In other words, users will continue to have high preferences on smart devices with large screen. Lots of articles have been published to introduce what phablets are and explain why they are useful and gain a huge amount of popularity. One of the reasons which supports this trend is that the phablet is able to fulfill people's desire: using a device that has both smart phone and tablet's strength. In comparison to other ordinary smart devices, the phablets' specifications are much better so that they can be used longer and faster in terms of the battery and advanced applications such as games, painting tools, etc. Moreover, because of the size of the device, users easily carry them in their pockets. In the modern society where people currently live in, they often finish their jobs while they're on transportation or on foot. Therefore, it seems to be natural that people required highly developed smart device with appropriate screen size and the phablets are the devices which satisfy their needs.

As both the penetration of smart devices and the use of large-screen smart devices have grown, the importance of the touch interface, the most typical input method for smart devices, has also increased. To support the statement, most of the children who had participated in the survey had positive opinions for using their fingers to touch the screen [2] andthere is a research trying to identify what the gestures are given users' inputs [3].A typical experiment that evaluates usability between pinch and force touchhad conducted with elderly people and the result showed they spent less time and had no difficultyin completing a task with force touch [4]. Althoughmost of them was not familiar with new touch technology and perhaps they are not willing to accept the new technology, youngergenerationtends to havea positive attitude on it when it gives them convenience. Therefore, a smartphone which has better touch interfacewill be dominant in mobile device industry.

In particular, the fact that smart devices with 3D-touch are able to perform more inputs than those based on 2D-touch and reduce input time makes users perform different tasks fast [5]. An experiment to compare which touch technology spends less time when people run a specific function in an application was conducted on YouTube and the result showed 3D-touch took much less time to complete the given missions. That implies 3D-touch interface has a huge advantage: using a function without running an application directly.

This trend is likely to increase the use of large screen smart devices based on 3D-touch technology. Because users tend to use smart devices with one hand, an increase in the screen size of smart devices is a strong factor that contributes to an increase in unstable touch area. It leads to a reasonable prediction that unstable touch area in 3D-touch interface is different from the area in 2D-touch interface. This comes from a touch process that includes not only the location of touching the screen, but also the touch force created by users' hand. Although there are researches that provide solutions in terms of how to minimize or overcome the unstable touch area [6], they do not show enough verification that same methods can be applied in the 3D-touch interface.

In this paper, we present an analysis of defining stable touch areas in two touch interfaces by distinguishing between unstable and stable touch areas through user experimentation and a future work to utilize the result. The analysis that we will present involves statistical methods in order to verify whether the results from two touch interfaces come from the same distribution.

Chapter 2 introduces tilting gesture, one of the ways to overcome unstable touch areas, and describes the result of implementation of tilting gesture and of the error rate in the 3D-touch interface to define stable touch area. Chapter 3 explores whether a smart device experiment with two touch technologies can calculate a stable touch area. In chapter 4, we verify whether the result values are able to distinguish between 2D-touch and 3D-touch interface, set the stable touch area on the 3D-touch interface, and discover the area where users have touch issues on the 3D-touch interface. Finally, chapter 5 shows conclusion and future research tasks.

2 Related Works

2.1 Tilting Gesture

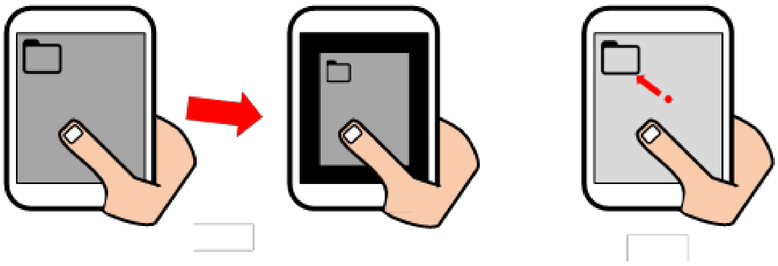


Fig. 1. Tilting gesture operation

As shown in Figure 1, when the user is using a smart device, the tilting gestures either reduce the whole screen by the certain proportion or create a temporary cursor in order for users to run an application or do other tasks. The tilting gesture is an interface method that can be used to overcome situations where using the device with one hand cannot reach all areas on the screen. Such tilting interface consists of three gestures: *tiltSlide, tiltReduction, and tiltCursor.*

TiltSlide makes the size of the screen smaller and located nearby an edge. The tilting technology installed in iphones called reachability is currently available, but it has the limitation of moving the screen downward only. However, the tiltSlide allows the screen to be moved to 8 directions, which is an advantage to use regardless of the size of the user's hand.

TiltReduction is a touch method by customizing the reduction ratio of the entire screen. The difference between tiltSlide and tiltReduction is that tiltSlide is a method of reducing the screen in a predetermined ratio. Therefore, if a user has significantly a small hand, then the adjusted screen by tiltSlide might not work. On the other hand, tiltReduction allows the user to adjust the ratio and the position where the adjusted screen will be located. The feature of the adjusted screen is basically a rectangle same as the size of the original screen.

Finally, **tiltCursor** is a way not to adjust the size of the screen or the location, but to create a cursor by a touch and move it to the desired location and execute functions. It shows a faster execution time in terms of simple touch operation, such as running apps or clicking links because the movement of the cursor is faster than the movement of the user's hand [7].

These tilting gesture interfaces should be created to complement a limited area that the user inputs. Therefore, it is necessary to apply the tilting gestures that can perform the operation to the remaining area excluding the limited area after the research on the user input limited space is preceded.

2.2 A study on the limited input space of users

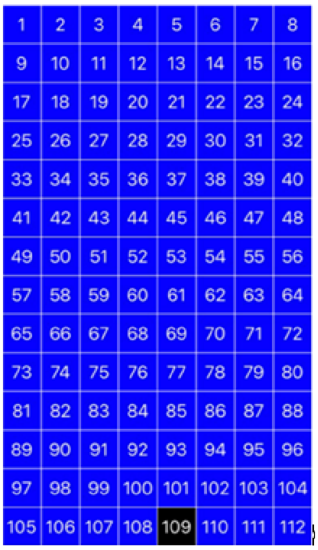
In Chang et al. [7], it divided Samsung Galaxy Note 3 into 112 buttons and measure task complete time and degree of inconvenience for each button to define a stable touch area for the 2D-touch interface. The reason to divide it into 112 buttons is that is the maximum number of stable touch buttons for a touch interface [8]. The result showed that the values of buttons, mainly around the center of the screen, were low, making them generally quick to perform and comfortable to perform. Furthermore, the average of work completion time and error rate of the three tilting gestures were experimented and when they compared with direct touch or 2D-touch, only tiltReduction showed the better performance than 2D-touch. The error rate of tiltSlide is the lowest among them and the tiltReduction has the highest error rate. Since all three gestures show lower error rates than 2D-touch without a significant difference, tiltReduction appears to be the fastest way for users to complete tasks because of the ability to customize the screen size.

Each method compares the task completion time of each button to 2D-touch, which shows that the tiltReduction is generally faster than 2D-touch, a considerable number of buttons experimented by tiltSlides and tiltCursors show the lower completion time than the 2D-touch. Therefore, tiltCursor and tiltSlide are not competitive compared to the 2D touch, and the tiltReduction seems to be the fastest and the most accurate way to use it among the other gestures. When using the 3D Touch interface, applying the tilting gesture is different from the 2D-touch interface, so it is difficult to apply it immediately. To be useable in the 3D-touch interface, a stable touch area must be defined first. To do this, we measured the error rate of the 3D-touch by dividing the screen of the smart device into 14 rows and 8 columns, respectively, with an iPhone 7 Plus, one of iphones capable of 3D-touch [9]. The experimental result shows that the error rates of the first row and the bottom two rows are higher than those of the other rows, and the error rate increases with the left column. However, since this method is to judge only based on rows and columns, it is necessary to divide into 112 zones as in the previous experiment to have more accurate results.

3 Experiment

3.1 App development for experiment

For the experiment on defining the tiltable area for 3D-Touch, we implemented the experimental application using the iPhone 7 Plus capable of 3D-touch. As shown in Figure 2, the experimental app divides the screen into 14 rows and 8 columns and constructs a button with a total of 112. The size of an icon on a smartphone is 8 mm x 8 mm, indicating a successful touch rate of more than 90% on 2D-touch. Each button provides color feedbacks (2D - Touch Pink, 3D - Touch Red) when applying 2D-touch and 3D-touch. The development environment was developed as shown in Table 1.



1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88
89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104
105	106	107	108	109	110	111	112

Fig. 2. Experimental application

Table 1. Development environment

	Smart device (iPhone 7P)	Development Computer (MacBook Pro 2013)
CPU(AP)	Apple A10 Fusion	Intel Core i7 (2.3GHz)
Memory	3GB	16G
OS	iOS 11.4.1	OS X 10.13.6
Language	Swift 4.1.3	Swift 4.1.3 (Xcode 9.4.1)
Display	5.5 inch	-
Touch Type	2D touch and 3D touch	-

3.2 Experimental Order

The experimenters, who participated in the experiment, are a total of 24 and all right-handed and are required to touch specific buttons when the number on each button turns into black in both 2D-touch and 3D-touch cases. Prior to the experiment, the subjects were required to perform each experiment 10 times in order to adapt to the experiment, thus recognizing the purpose of the experiment.

The subjects were asked to position the thumb at the starting point between the two buttons, 52 and 53, and to press the required button on the screen. A total of 10 repetitions were performed for all buttons per subject. The satisfaction level for each input with 2D-touch and 3D-touch was from 1 point (no inconvenience at all) to 5 points (very uncomfortable). If an input for a required button is not made, an error and the response time are measured until the input for each button is measured.

3.3 Experimental analysis method

After obtaining the results of the experiment in three fields, we first compare the data of each field by statistical method to confirm whether to apply the tilting gesture in 2D Touch and 3D Touch interface differently. First, we perform a Shapiro-Wilk test to confirm that the distribution of the data from the two touch is a normal distribution.

Since the data generated by each button is then the result of an experiment with the same mobile device, we consider the data as a paired data. Given the p-value calculated by the Shapiro-Wilk test, if the p-value is greater than a significance level (alpha = 0.5), we use Paired student's t-test, and if not, then we test the data with Wilcoxon signed-rank test.

Prior to the analysis, we adjust the touch satisfaction with min-max normalization, from 0 (most uncomfortable) to 1 (most comfortable). Likewise, we change the error rate to accuracy, meaning that 0 is the least accurate and 1 is the most accurate.

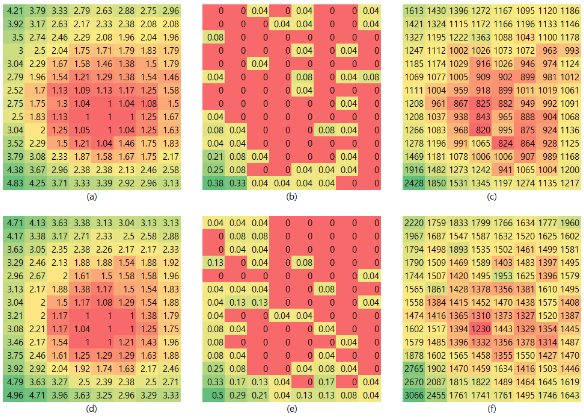
The average of touch satisfaction and accuracy for each button is determined by the harmonic mean, because we consider the stability as the average of the two factors, the touch satisfaction and the accuracy, in this experiment. If either one of two factors is significantly low, the button cannot be stable. Therefore, it is necessary to obtain an average that is not distorted by an extreme value.

Based on these values, the following conditions must be satisfied to set the tiltable area including the stable touch area in the touch interface. 1) The tiltable area should be rectangular or square. 2) All values within the set rectangle must be above a given threshold. 3) If two or more rectangles appear to have the same number of buttons, we compare the values of the remaining buttons except for the common part and choose a rectangle that contains more high values. We also define areas where 3D Touch is difficult by finding values that are significantly lower than others.

4 Experiment Result

As we can see in Figure 3, two datasets in the response time for 2D-touch and 3D-touch is significantly different. Unlike 2D Touch, where buttons surrounding the center of the screen response within 1,000 milliseconds, buttons in 3D-touch spend at least 1230 milliseconds to answer. This seems to be a nature characteristic of the 3D-touch interface, which requires more than 200 milliseconds to be inputted.

Overall, the touch satisfaction shows that the values at the four edges tend to be higher than the center, and the buttons at the vertex have higher numbers than the others. Direct comparison shows that the portion with the highest 2D-touch satisfaction is slightly wider than the portion with the 3D-touch. The error rate reveals the tendency that the value at the southwest corner of both touch technologies is the largest, and the part with low error rate is more widely distributed in the 2D-touch interface. Table 2 and 3shows the results obtained by using the statistical methods mentioned above for more accurate analysis.



4.21	3.79	3.33	2.79	2.63	2.88	2.75	2.96
3.92	3.17	2.63	2.17	2.33	2.38	2.08	2.08
3.5	2.74	2.46	2.29	2.08	1.96	2.04	1.96
3	2.5	2.04	1.75	1.71	1.79	1.83	1.79
3.04	2.29	1.67	1.58	1.46	1.38	1.5	1.79
2.79	1.96	1.54	1.21	1.29	1.38	1.54	1.46
2.52	1.7	1.13	1.09	1.13	1.17	1.25	1.58
2.75	1.75	1.3	1.04	1	1.04	1.08	1.5
2.5	1.83	1.13	1	1	1	1.25	1.67
3.04	2	1.25	1.05	1	1.04	1.25	1.63
3.52	2.29	1.5	1.21	1.04	1.46	1.75	1.83
3.79	3.08	2.33	1.87	1.58	1.67	1.75	2.17
4.38	3.67	2.96	2.38	2.38	2.13	2.46	2.58
4.83	4.25	3.71	3.33	3.39	2.92	2.96	3.13

(a)

0	0	0.04	0	0.04	0	0	0.04
0	0	0	0.04	0.04	0	0	0.04
0.08	0	0	0	0	0	0	0
0	0	0	0	0.04	0	0.04	0
0	0	0.04	0	0	0	0	0
0.04	0	0	0	0.08	0	0.04	0.08
0	0	0	0	0.04	0	0	0
0	0	0	0	0	0	0.04	0
0.04	0	0	0	0	0	0	0
0.08	0.04	0	0	0	0.08	0.04	0
0.04	0.04	0	0	0	0	0	0
0.21	0.08	0.04	0	0	0	0.04	0
0.25	0.08	0	0	0.04	0.04	0.04	0
0.38	0.33	0.04	0.04	0.04	0.04	0	0

(b)

1613	1430	1396	1272	1167	1095	1120	1186
1421	1324	1115	1172	1166	1196	1133	1146
1327	1195	1222	1363	1088	1043	1100	1178
1247	1112	1002	1026	1073	1072	963	993
1185	1174	1029	916	1026	946	974	1124
1069	1077	1005	909	902	899	981	1012
1111	1004	959	918	899	1011	1019	1061
1208	961	867	825	882	949	992	1091
1208	1037	938	843	965	888	904	1068
1266	1083	968	820	995	875	924	1136
1278	1196	991	1065	824	864	928	1125
1469	1181	1078	1006	1006	907	989	1168
1916	1482	1273	1242	941	1065	1004	1200
2428	1850	1531	1345	1197	1274	1135	1217

(c)

4.71	4.13	3.63	3.38	3.13	3.04	3.13	3.13
4.17	3.38	3.17	2.71	2.33	2.5	2.58	2.88
3.63	3.05	2.35	2.38	2.26	2.17	2.17	2.33
3.29	2.46	2.13	1.88	1.88	1.54	1.88	1.92
2.96	2.67	2	1.61	1.5	1.58	1.58	1.96
3.13	2.17	1.88	1.38	1.17	1.5	1.54	1.83
3.04	2	1.5	1.17	1.08	1.29	1.54	1.88
3.21	2	1.17	1	1	1	1.38	1.79
3.08	2.21	1.17	1.04	1	1	1.25	1.75
3.46	2.17	1.54	1	1	1.21	1.43	1.96
3.75	2.46	1.61	1.25	1.29	1.29	1.63	1.88
3.92	2.92	2.04	1.92	1.74	1.63	2.17	2.46
4.79	3.63	3.27	2.5	2.39	2.38	2.5	2.71
4.96	4.71	3.96	3.63	3.25	2.96	3.29	3.33

(d)

0.04	0.04	0.04	0	0	0	0	0
0	0.08	0.08	0	0	0	0	0
0	0.04	0.04	0	0	0	0	0
0.13	0	0.04	0	0.08	0	0	0
0	0	0	0	0	0	0	0.04
0.04	0.04	0.04	0	0	0.08	0	0
0.04	0.13	0.13	0	0	0	0	0.04
0.04	0	0	0.04	0.04	0	0	0
0.08	0.04	0	0	0	0.04	0	0
0.08	0	0	0	0	0	0.04	0
0.08	0.04	0	0	0	0.04	0	0
0.25	0.08	0	0	0.04	0.08	0.04	0
0.33	0.17	0.13	0.04	0	0.17	0	0.04
0.5	0.29	0.21	0.04	0.13	0.13	0.08	0.04

(e)

2220	1759	1833	1799	1766	1634	1777	1960
1967	1687	1547	1587	1632	1520	1625	1602
1794	1498	1893	1535	1502	1461	1499	1581
1790	1509	1469	1589	1403	1483	1397	1495
1744	1507	1420	1495	1953	1625	1396	1579
1565	1861	1428	1378	1356	1381	1610	1495
1558	1384	1415	1452	1470	1438	1575	1408
1474	1416	1365	1310	1373	1327	1520	1387
1602	1517	1394	1230	1443	1329	1354	1445
1579	1485	1396	1332	1356	1378	1314	1487
1878	1602	1565	1458	1355	1550	1427	1470
2765	1902	1470	1459	1634	1416	1503	1446
2670	2087	1815	1822	1489	1464	1645	1619
3066	2455	1761	1741	1761	1495	1746	1643

(f)

Fig. 3. (a) 2D-touch Satisfaction, (b) the error rate of 2D-touch, (c) 2D-touch response time, (d) 3D-touch Satisfaction, (e) the error rate of 3D-touch, and (f) 3D-touch response time

Table 2. Statistical analysis result for comfort rate

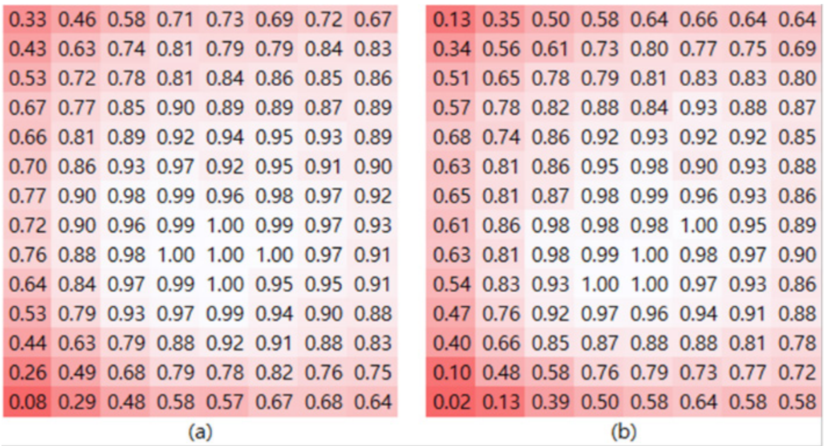
Statistical Method	Types of Touch Screen Technology	Comfort rate	
		Test statistic	*P-value*
Shapiro-Wilk	2D-Touch	0.9375	5.33x10^{-5}
	3D-Touch	0.94701	0.0002271
Wilcoxon Signed-rank	2D-Touch 3D-Touch	484.5	2.025x10^{-13}

Table 3. Statistical analysis result for error rate

Statistical Method	Types of Touch Screen Technology	Error rate	
		Test statistic	*P-value*
Shapiro-Wilk	2D-Touch	0.45404	<2.2x10^{-16}
	3D-Touch	0.58199	2.398x10^{-16}
Wilcoxon Signed-rank	2D-Touch 3D-Touch	481	0.001672+

As shown in Table 2 and 3, 2D-touch and 3D-touch can be identified with the touch satisfaction and the error rate. Since the classification in response time is quite successful with direct comparison, we conclude that we do not need statistical method. First, both 2D and 3D Touch data distributions do not seem to be the normal distribution as the Shapiro-Wilk test showed significantly small p-value in the touch satisfaction and the error rate. Since the Wilcoxon signed-rank test shows that the p-value was close to zero, this indicates that the values derived from each of the buttons on the two interfaces were not the same. The two interfaces also show that they can be distinguished based on the touch satisfaction level or error rate mentioned earlier.

Therefore, the criteria for setting the tiltable areas of the 3D-touch interface should not be the same as the 2D-touch interface.



0.33	0.46	0.58	0.71	0.73	0.69	0.72	0.67
0.43	0.63	0.74	0.81	0.79	0.79	0.84	0.83
0.53	0.72	0.78	0.81	0.84	0.86	0.85	0.86
0.67	0.77	0.85	0.90	0.89	0.89	0.87	0.89
0.66	0.81	0.89	0.92	0.94	0.95	0.93	0.89
0.70	0.86	0.93	0.97	0.92	0.95	0.91	0.90
0.77	0.90	0.98	0.99	0.96	0.98	0.97	0.92
0.72	0.90	0.96	0.99	1.00	0.99	0.97	0.93
0.76	0.88	0.98	1.00	1.00	1.00	0.97	0.91
0.64	0.84	0.97	0.99	1.00	0.95	0.95	0.91
0.53	0.79	0.93	0.97	0.99	0.94	0.90	0.88
0.44	0.63	0.79	0.88	0.92	0.91	0.88	0.83
0.26	0.49	0.68	0.79	0.78	0.82	0.76	0.75
0.08	0.29	0.48	0.58	0.57	0.67	0.68	0.64

(a)

0.13	0.35	0.50	0.58	0.64	0.66	0.64	0.64
0.34	0.56	0.61	0.73	0.80	0.77	0.75	0.69
0.51	0.65	0.78	0.79	0.81	0.83	0.83	0.80
0.57	0.78	0.82	0.88	0.84	0.93	0.88	0.87
0.68	0.74	0.86	0.92	0.93	0.92	0.92	0.85
0.63	0.81	0.86	0.95	0.98	0.90	0.93	0.88
0.65	0.81	0.87	0.98	0.99	0.96	0.93	0.86
0.61	0.86	0.98	0.98	0.98	1.00	0.95	0.89
0.63	0.81	0.98	0.99	1.00	0.98	0.97	0.90
0.54	0.83	0.93	1.00	1.00	0.97	0.93	0.86
0.47	0.76	0.92	0.97	0.96	0.94	0.91	0.88
0.40	0.66	0.85	0.87	0.88	0.88	0.81	0.78
0.10	0.48	0.58	0.76	0.79	0.73	0.77	0.72
0.02	0.13	0.39	0.50	0.58	0.64	0.58	0.58

(b)

Fig. 4. Average values of 2D-touch (a) and 3D-touch (b) in 112 buttons

The average values of both touch (Figure 4), calculated based on the touch satisfaction and the error rate, shows that both 2D-Touch and 3D-Touch have higher values near the center of the screen implying that stable touch area seems to be located within the white colored buttons. The number of the average value in each button that satisfies each threshold in 2D-touch appears higher than in 3D-touch. Another fact that we want to point out is the results of the 3D-touch experiment show that the buttons on top-left corner and bottom-left corner are significantly lower than in 2D-touch. This implies that the characteristic of 3D-touch, measuring the touch force, affects those areas where the user's finger has a trouble to touch.

Finally, the tiltable area in 2D-touch based on a threshold is always wider, except for a thresh value of 0.75, when we create a rectangle according to the conditions mentioned above. In cases of 0.90 and 0.95, the left and right sides of the area that satisfies the given threshold is slightly wider in 2D-touch, and the upper and lower sides of the area is wider in 3D-touch. Because the 2D-touch interface receives only inputs of the position where a finger touches the screen without giving a force, moving fingers left or right is more comfortable than moving north and south. In case of setting the tiltable area for the 3D-touch interface which receives a touch force, a rectangular shape that has a longer height than width seems to be more appropriate.

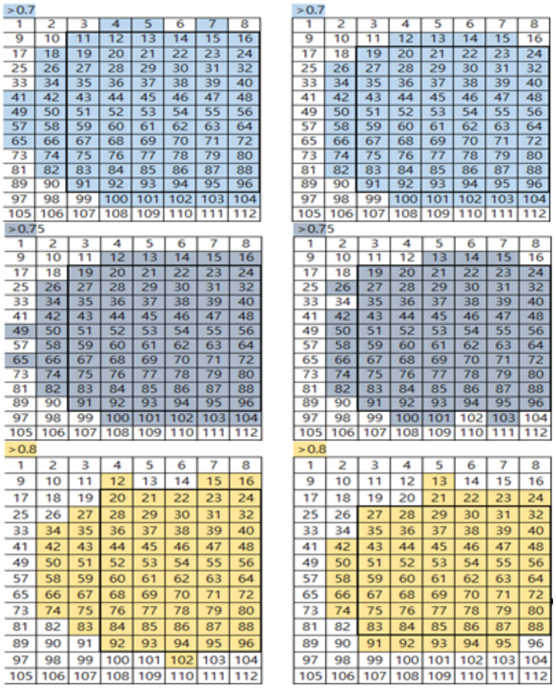


Fig. 5. Tiltablearea based onaverage values in 2D-touch (left) and 3D-touch (right) given a threshold value of 0.7, 0.75 and 0.8

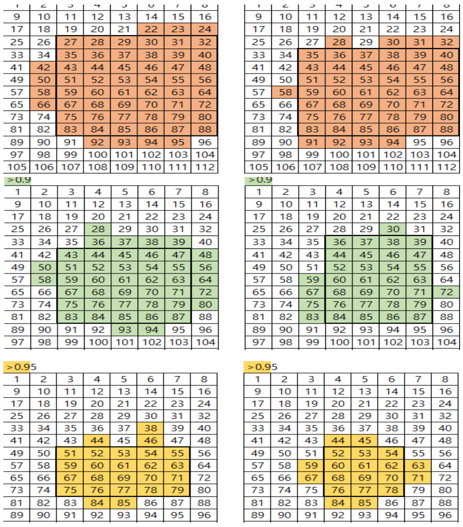


Fig. 6. Tiltable area based on average values in 2D-touch (left) and 3D-touch (right) given a threshold value of 0.85, 0.9 and 0.95

5 Discussion

Figure 4, Figure 5 and Figure 6 clearly supports why we should apply a different criterion to 3D-touch interface. Figure 5 shows when we make a tiltable area in 3D-touch interface, the size is mostly smaller in each condition. In Chapter 4, we mentioned what the differences were by comparing the results from both touch interfaces and conditions for the tiltable area: a value in each button should be greater than the given threshold in this case.

The experiment that we proceeded was conducted with iphone7 plus as above, the result from other devices such as Galaxy or other products might be different due to the method that receives the strength of users' touch in 3D-touch interface. However, it is reasonable to think the method among companies is similar, so we can assume that results on other smart devices will converge to ours. Another component that may affect our experiment result is a gender. Since we did not analyze the data separately, the criteria for setting tiltable area for both men and women is currently undefined. Although there are some factors that contribute the fact that the average of both genders' hand size is not same so that it might lead to the different result, calculating the mean for all genders with a considerable amount of data will not be a huge issue to define the tiltable area.

In terms of hands, all experimenters are right-handed and grabbed the lower right corner. First, if we had left-handed people for the experiment, the tiltable area would have been a different shape on each given threshold. As shown in Figure 5 and Figure 6, the tiltable area tend to be close to the right side of the screen for the right-handed people when the threshold is less than 0.95 for both 2D-touch and 3D-touch. Thus, if we define a tiltable area for left-handed, then the area may be opposite: the area is close to the left side of the screen. Second, we asked all participants to hold the bottom of the device because that is what users feels comfortable when they grab their phone. Even though some users grab a different part of their device, based on user's general habit, we conducted the experiment in such way.

When the tilting gesture is applied in practical world, all functions and application shown in the screen must be adjusted. Because the screen consists of 112 buttons, 14 rows and 8 columns, whatever tilting area we choose, it is desirable that everything including the whole screen that exists in the screen keep the ratio for convenience. In other words, for future experiments, we have to make sure tasks, which we want to complete work appropriately in the tiltable area in each case and discover which tiltable area should be used.

6 Conclusion

This paper defined stable touch areas for both 2D-touch and 3D-touch by constructing 112 buttons that consists of 14 rows and 8 columns first and calculate the mean of normalizing the rate of touch satisfaction and accuracy, a substitution for the error rate, based on 3 criteria that we measured. Because the screen of the smart device tends to be a rectangle, it is reasonable that we find a rectangle that buttons in that area should have a greater value than a given threshold. We can conclude that the tiltable area shrunk towards the center of the mobile screen as the threshold increases. In addition, in terms of defining the tiltable areas, the areas in 3D-touch is always different from those in 2D-touch and the number of buttons in the areas is smaller. Moreover, we discovered areas where 3D-touch is particularly more difficult than 2D-touch by comparing two means from each button to focus on applying tilting gesture to the 3D-touch interface.

In the future, we concentrate on verifying this result when we perform a tilting gesture experiment applying 3D-touch technology to a mobile device by comparing the result from 2D-touch technology. In addition, we identify whether the screen size of mobile devices affects the performance of the tilting gesture on the 3D-touch interface.

7 Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2017R1D1A1B03032855)

8 References

[1] R. Williams, "IDC: Phablets to overtake smaller smartphones by 2019", Mobile Marketer, 2017. [Online]. Available: https://www.mobilemarketer.com/news/idc-phablets-to-overtake-smaller-smartphones-by-2019/511962/. [Accessed: 30-Aug-2018].

[2] R. Kraleva, "Designing an Interface For a Mobile Application Based on Children's Opinion", International Journal of Interactive Mobile Technologies, vol. 11, no. 1, pp.53-70, 2017.http://online-journals.org/index.php/i-jim/article/view/6099/4250 https://doi.org/10.3991/ijim.v11i1.6099

[3] S. A. Al-Showarah, "The Effectiveness of Dynamic Features of Finger Based Gestures on Smartphones' Touchscreens for User Identification", International Journal of Interactive Mobile Technologies, vol. 11, no. 1, pp.133-142, 2017.http://online-journals.org/index.php/i-im/article/view/6368/4251https://doi.org/10.3991/ijim.v11i1.6368

[4] P. Murano and R. M. D. Ali, "A Preliminary Investigation into the Usability of Force Touch Interaction in a Mobile Context", International Journal of Interactive Mobile Technologies, vol. 12, no. 5, pp.169-181. 2018. http://online-journals.org/index.php/i-jim/article/view/9208/5176 https://doi.org/10.3991/ijim.v12i5.9208

[5] C. Nam and D. Shin, "Force-touch measurement methodology based on user experience", International Journal of Distributed Sensor Networks, vol. 14, no. 2, 2018. https://journals.sagepub.com/doi/full/10.1177/1550147718767794

[6] S. Kim, J. Yu and G. Lee, "Interaction Techniques for Unreachable Objects on the touch screen: an experimental evaluation of three strategies", Proceedings of the 24thAustralian Computer-Human Interaction Conference, Melbourne, Australia, November 26-30, pp. 295-298, 2012. https://dl.acm.org/citation.cfm?id=2414585

[7] Y. Chang, S. L'Yi, K. Koh, and J. Seo, "Understanding Users' Touch Behavior on Large Mobile Touch-Screens and Assisted Targeting by Tilting Gesture", Proceedings of the

33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Republic of Korea, April 18-23, pp. 1499-1508, 2015. https://dl.acm.org/citation.cfm?id=2702425

[8] P. Parhi, A. K. Karlson and B. B. Bederson, "Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices", Proc. of the 8th Conference on Human-computer Interaction with Mobile Devices and Services, Helsinki, Finland, September 12-15, pp.203-210, 2006. https://dl.acm.org/citation.cfm?id=1152260

[9] J. Lee, C. Nam, D. Shin, "A Study on 3D-Touch Usability on Large Screen Touch Device", Proceedings of ICCAS 2017, Jeju, Korea, October 18-21, pp.1281-1282, 2017.

9 Authors

YounghoonSeo is a student at Sungkyunkwan University and currently studying for a M.S. in Convergence Institute for Intelligence and Informatics department. His research interests are reinforcement learning and artificial neural network.

Dongryeol Shinis a professor of Software at SungkyunkwanUniversity since 1994. He received the B.S. degree in Electronic Engineering at SungkyunkwanUniversity in 1980, the M.S degree. in Electrical and Electronic Engineering at Korea Advanced Science and Technology in 1982, and the Ph.D. degree in Electrical Computer Engineering at Georgia Institute of Technology in 1992. His current research interest is sensor network.

Choonsung Namis a research professor at Sungkyunkwan University and had worked for VANET and UI/UX for 3D-touch at Yonsei University as a Post-doc. He received B.S, M.S., and Ph.D. degrees in Software from SangMyung University, SoongSil University, 2007, and in Computer Science from Sunkyunkwan University, 2011, in Korea. His research interests include VANET, Sensor Networks, Drone, IoT, UI/UX for 3D-Touch and so on.

Article submitted 16 October 2018. Resubmitted14 January 2019. Final acceptance 19 January 2019. Final version published as submitted by the authors.