Two-stage Recognition of Raw Acceleration Signals for 3-D Gesture-Understanding Cell Phones

Sung-Jung Cho, Eunseok Choi, Won-Chul Bang, Jing Yang, Junil Sohn, Dong Yoon Kim, Young-Bum Lee and Sangryong Kim
Interaction Lab.
Samsung Advanced Institute of Technology
{sung-jung.cho, eunseok.choi, wc.bang, jing.yang, ji.sohn, kdy2891, srkim}@samsung.com

Abstract
As many functionalities like cameras and MP3 players are converged to cell phones, more intuitive interaction methods are essential beyond tiny keypads. In this paper, we present gesture-based interactions and their two-stage recognition algorithm. Acceleration signals are generated from accelerometer. At the first stage, they are hierarchically modelled and matched as basic component and their relationships by Bayesian networks. At the second stage, they are further classified by SVMs for resolving confusing pairs. Our system showed enough recognition performance for commercialization; with 100 novice users, the average recognition rate was 96.9% on 11 gestures (digits 1-9, O, X). The algorithms have been adopted in the world-first gesture-recognizing Samsung cell phones since 2005.

Keywords: 3-D gesture recognition, acceleration, Bayesian networks, support vector machines, sensor-based interaction

1. INTRODUCTION
As the role of cell phones has evolved from mere voice communication devices to our daily life assistants, they have employed more functionalities like cameras, MP3, and web browsing. Even though the evolution enables users to enjoy various functions at any time, it incurs the usage problem of controlling many functions with tiny screen and keypads. Therefore, intuitive and interesting interaction methods are essential in mobile devices.

These days, a new kind of interaction technology that understands users’ movement has emerged due to the rapid development of sensor technology. An accelerometer measures the amount of acceleration of a device in motion. Analysis of acceleration signals enables three kinds of gesture interaction methods: tilt detection, shake detection and gesture recognition [1,2].

The tilt detection algorithm interprets the posture of a device. When a user holds it in a static posture, its tilt angle is calculated by measuring the ratio of gravity components in tri-axis. It is used for moving cursors in a menu tree or virtual objects [3,4]. By combining tilt-based input and RFID-based object identification, information on physical objects can be browsed [5].

The shake detection algorithm interprets occurrences of users’ shake movement. When a user shakes a phone, acceleration signals in a time interval are analyzed about whether they exceed threshold values. It is used for counting the number of walking steps in Fujitsu’s cell phone F6721 and Pantech’s PH-S6500 [3]. Also, shaking patterns are used for identifying users and devices [6,7].

The gesture recognition algorithm interprets dynamic movement patterns in the 3-D space. When a user draws a trajectory in the air for inputting commands or characters, the relationship between acceleration signals over the whole input is analyzed. We proposed a remote controller prototype, Magic Wand, for controlling TVs by gestures in the air [8-9] with accelerometers and gyroscopes. The 3-D trajectories are estimated by employing inertial navigation theory and then classified. Mäntyjärvi et al. also published a gesture-interactive remote controller for controlling DVD players with an accelerometer for recognizing eight gestures [10].

The goal of our research is to commercialize gesture-understanding technologies in cell phones for supporting interesting and intuitive interaction experiences to users. The previous researches have following limitations for the goal. First, tilt-based input is analogous to four-direction keys so that users’ interests and curiosity are not so large. Second, the previous shaking detection algorithms do not handle applications with real-time response requirements. Third, previous gesture recognition algorithms are not of commercial quality because recognition approach after trajectory estimation requires gyroscopes which are not still fitted well to cell phones in terms of size and cost [8]. Also Mäntyjärvi et al.’s work lacks in the user-independent recognition capability [10].

In this paper, we present two-stage gesture recognition system and their applications in the world-first commercialized gesture interactive cell phone (Samsung SCH-S310 and E760 released in 2005) [11,12]. The gesture recognition algorithm enables the inputting of characters and symbols by drawing them in the air as an intuitive interaction. Acceleration signals generated according to users’ motion are hierarchically modelled and matched as basic component and their relationships by Bayesian networks. Then they are further classified by SVMs for resolving confusing pairs.
The rest of this paper is organized as follows. Section 2 describes the overview of the gesture interactive cell phone. Section 3 presents the gesture recognition algorithm. Section 4 shows experimental results and Section 5 concludes the paper.

2. GESTURE INTERACTIVE CELL PHONES

Figure 1 shows the overview of gesture interactive cell phones. When a user draws gestures, the movement is sensed by an accelerometer. Then the acceleration signal is processed and normalized. It is then classified into a gesture by the gesture recognition algorithm. Finally, the corresponding function is executed and shown to users.

2.1. Gesture-based Interaction Applications

Among the three interaction categories, we choose real-time shake-based interaction and gesture-based interaction for supporting intuitive interface and entertainment applications. They are targeted for young generations from 10’s to 30’s who are very sensitive to new trends. Samsung cell phone design groups utilize their expertise in developing application ideas with us.

The real-time shake-based interaction supports mainly entertainment applications. Figure 2 shows screen shots of Dices and Random balls. In Dices, the dices start rolling when the cell phone is shaken and stop when not shaken. In Random balls, the balls are rotating when shaken and randomly selected. The games look very realistic because they resemble our activity of shaking dices and balls in the real world. The shaking movement is also used for generating music: musical instrument sound like drum and portion of songs are played to the shaken time.

The gesture-based interaction supports command input for speed dialing, gesture-to-sound generation, song navigation and message deletion (Figure 3). By using gestures, user does not need to pay attention to keypads, which may be useful for the blind. Also it is convenient for slide-up style cell phones whose keypads are hided by the upper screen.

In speed dialing, a user makes a phone call to the registered phone number by drawing digit shapes in the air. In gesture-to-sound generation, the user draws O or X on the air. Then, the sound corresponding to the gesture is played such as ‘I love you’ or ‘Oh, No~’. It is used for expressing the user’s emotion in interesting way.

In song navigation, the user moves to the next or the previous song in the song list by shaking it rightward or leftward while listening to music. In message deletion, the user deletes messages like advertisement by shaking the phone vertically twice.

Table 1 shows the phone functions and their gesture shapes [2]. In idle phone context where the phone waits for phone calls, a user can dial phone number or generate gesture sounds by drawing shapes of {1-9, O, X}. In MP3 played context, he can select the previous or the next song by gestures. In message received state, he can delete newly arrived message by gestures.

<table>
<thead>
<tr>
<th>Context</th>
<th>Function</th>
<th>Gesture shapes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle</td>
<td>Speed dialing</td>
<td>1 2 3 4 5 6 7 8 9</td>
</tr>
<tr>
<td></td>
<td>Gesture-to-sound</td>
<td>O X</td>
</tr>
<tr>
<td>MP3 played</td>
<td>Song navigation</td>
<td></td>
</tr>
<tr>
<td>Message received</td>
<td>Message deletion</td>
<td>N</td>
</tr>
</tbody>
</table>

Table 1. Gesture phone functions and their corresponding shapes

2.2. Hardware Components

The gesture recognition capability requires additional hardware components: a gesture button, analog-to-digital converter (ADC), and tri-axis accelerometers. The gesture button is used for indicating the start and the ending time of gesture input. The ADC digitizes acceleration signals for digital processing. The accelerometer generates acceleration signals while the button is pressed. We use the accelerometer KXM 52 from Kionix [13] because it senses the range of +- 2G (gravity force) acceleration at 100 Hz sampling frequency which is enough for sensing accelerations due to hand movements.

Fig. 1. Overview of Gesture Interactive Cell Phone

Fig. 2. Game applications: Dices and Random balls

Fig. 3. Gesture applications: speed dialing, sound generation, MP3 control, and message deletion
3. Gesture recognition algorithm

The recognition framework consists of five steps: calibration, preprocessing, feature extraction, recognition and confusing pair discrimination (Figure 4).

When a user presses the gesture activation button, the calibration module converts the digital sensor output values in the range of 0 to 255 into physical acceleration values m/s<sup>2</sup> by rescaling and translating them. The preprocessing step normalizes acceleration signal variations like slow or fast writing. The feature extraction step finds feature points and divides acceleration signals into primitives at those feature points. The recognition step matches the gesture input with Bayesian network-based gesture models and finds the most probable gesture model given the input. Finally, the confusing pair discrimination step is invoked for further discriminating frequently confusing gestures.

\[ A_2(t) = A_1(t)/\|A_1\| \]  

- Gaussian Smoothing

The obtained tri-axis accelerations contain measurement noises and a user’s unintended hand trembling. Hence, it is necessary to get rid of such noises for extracting reliable features.

1-D Gaussian smoothing is adopted for each axis with Gaussian kernel of zero-mean. The convolution weight for smoothing is given as

\[ G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \]  

where \( \sigma \) is the standard deviation.

The data after Gaussian smoothing is given as follows:

\[ A_3(t) = \sum_{i=t-k}^{t+k} G(i-t)A_2(t) \]  

- Resampling

Because acceleration signals are sampled in equal-time interval, the number of sampled points is variable according to the gesture input speed; the fast moving interval has small number of points and vice versa. Therefore, the writing speed should be normalized.

Our approach is to resample the signals to have same lengths in the acceleration-length space. Then any neighboring two resampled points have same distance in the space. The length of acceleration signals from time 1 to T in the original space is represented as follows:

\[ R_i^T(A) = \sum_{t=1}^{T} \|A_3(t) - A_3(t-1)\| \]  

The new points are resampled with following constraints:

\[ A_i(t) = \{P(i) \mid \text{on trajecory} \{A_3(1), ..., A_3(T)\}, \]  

\[ s.t. \ R_i^T(A_i) = U, \text{and} \ P(h) = A_i(t-1) \} \]  

where \( U \) denotes predetermined unit length which is determined by experiment.

3.1. Preprocessing

The preprocessing step consists of four sub steps: motion-area detection, normalization, Gaussian smoothing and resampling.

- Motion-area detection

It detects the interval of a gesture motion in time. By utilizing acceleration signals within the motion interval, a gesture is recognized regardless of whether users make pause during gesture input. The signals in motion interval are different from those in pause interval in that their variances are significantly larger. A sample point \( A(t) = [a_x(t), a_y(t), a_z(t)] \) is classified into motion area point by comparing it with \( K \)-previous samples:

\[ \exists_{i \in \{1, ..., K\}} | A(t) - A(t-k) | \geq \text{threshold} \]  

(1)

The motion area points in neighbor are merged consecutively. If several motion area groups are found, then the biggest motion area is selected.

- Norm normalization

The sensed acceleration signal contains not only the gesture movement acceleration but also earth gravity. The gravity amounts are different according to the posture of the sensor in the 3D space. Also, the gesture acceleration signal size changes according to the writing force. Those variations are normalized as follows.

The gravity components are approximately removed by subtracting the mean of accelerations at each time.

\[ A_1(t) = A(t) - \overline{A} \]  

Writing force is approximately represented by the norm of tri-axis acceleration signals. Therefore, it is normalized by making the mean of norm as one:

\[ A_2(t) = A_1(t)/\|A_1\| \]  

3.2. Feature Point Extraction

In the proposed gesture recognition system, primitives need to be extracted as basic modeling units. Here, a primitive means a portion of acceleration signals whose values increase or decrease monotonously within the interval. The boundary between successive primitives has the property that the signal value has local maximal or minimal value. A point is classified as local minima or maxima if it satisfies one of following conditions:

\[ \max_{x,y,z} A_x(t) \geq \forall_{x \neq k} A_x(t \pm k) \quad \forall_{x \in \{x, y, z\}} \]  

\[ \min_{x,y,z} A_x(t) \leq \forall_{x \neq k} A_x(t \pm k) \quad \forall_{x \in \{x, y, z\}} \]  

For the measured acceleration signals, the number of extracted local maximal or minimal points is generally excessive than the number of proper primitive boundaries. Therefore, they are filtered with the conditions that primitives should have significantly large
lengths and the local minima and maxima point should have sufficiently small and large value respectively.

3.3. Recognition Algorithms

3.3.1. Bayesian network modeling of relationships

A Bayesian network is a directed acyclic graph (DAG) whose nodes represent random variables and whose arcs relationships between them [14].

In this paper, the relationship between random variables are represented by the conditional Gaussian distribution. When a multivariate random variable $X$ depends on $X_1, \cdots, X_n$, the conditional probability distribution is given as follows:

$$P(X = x | X = x_1, \cdots, X_n = x_n) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-u)^T \Sigma^{-1} (x-u)\right]$$

The mean $\mu$ is determined from the linear weight sum of dependant variables $Z = [x_1^T, \cdots, x_n^T, 1]^T$ as follows:

$$u = Wz^T$$

where $W$ is a $d \times k$ linear regression matrix, $d$ and $k$ are the dimension of $X$ and $Z$ respectively.

3.3.2. Gesture model

A gesture is represented hierarchically by modeling its primitives and relationships among the primitives. In the first level, a gesture model is composed of primitive models and their relationships. A primitive represents a portion of acceleration signals whose values increase or decrease monotonously. In the second level, a primitive model is composed of point models and their relationships [15,16]. Finally, a point model is represented as the dependency of a mid point from two end points of a primitive. Figure 5 shows an example of recursive construction of a primitive model.

![Fig. 5. Example of recursive construction of a primitive model](image)

A gesture model is constructed by concatenating primitive models according to their input order and specifying inter-primitive relationships (IPR). IPRs are represented by dependencies among primitive boundary points. Figure 6 shows a Bayesian network based gesture model with $N$ primitives and the recursion depth $d = 2$. $EP_i$’s are the primitive boundary point models and $IP_{i,j}$’s are the internal point models of the $i$ -th primitive. The right end point of the previous primitive is shared with the left one of the following primitive. IPRs are represented by the arcs between $EP_i$’s, and WPRs are represented by the incoming arcs to $IP_{i,j}$’s.

![Fig. 6. Gesture model with N primitives and the primitive recursion depth of 2](image)

3.3.3. Matching algorithm

Each gesture class $m$ has a corresponding gesture model $\lambda_m$. A gesture input, a tri-axial acceleration sequence of $A(1), \cdots, A(T)$, is recognized by finding the gesture model $\lambda$ which produces the highest model likelihood as follows:

$$\lambda = \arg\max_m P(\lambda_m | A(1), \cdots, A(T))$$

The model likelihood is calculated by matching primitive internal point models ($IP_i$’s) and primitive end point models ($EP_i$’s) of gesture models (Figure 6) with the input point sequence. Because primitive boundaries are not explicitly specified in the input acceleration sequence, all the possible primitive segmentations should be searched. After a gesture input is segmented into primitives, primitive end points are matched to $EP_i$’s. Then each primitive instance is recursively matched to mid points $IP_i$’s. When a gesture model $G$ with $N$ basic primitive models matches the input
points $A(1), \cdots, A(T)$, and one stroke segmentation instance is denoted as $\mathcal{Y} = (t_0, t_1, \cdots, t_N)$, $t_0 = 1 \leq t_1, \cdots, \leq t_N = T$ and the whole set as $\Gamma$, the likelihood is calculated as follows:

$$L = R_1^T(A) = \sum_{i=2}^{T} \left\| A_3(i) - A_3(i-1) \right\|$$  \hspace{1cm} (13)

The new points are resampled with following constraints:

$$A^+(t) = \{ P(i) \ \text{on trajectory} \ A_3(1), \cdots, A_3(T), \ s.t. \ \gamma_k(A_j) = L / M, \ \text{and} \ P(h) = A_3(i-1) \}$$  \hspace{1cm} (14)

For SVM features, combination of $X$, $Y$, and $Z$-axis values and their norm values are used.

4. EXPERIMENTAL RESULTS

4.1. Gesture Shapes

Table 1 shows gesture shapes for all the applications. The shapes should be easy to learn and remember. In gesture-to-sound, $O$ denotes positive sound because it usually means good or OK. $X$ denotes negative sound because it usually means bad or not OK. In song navigation, ‘>’ shape is mapped to the next song title and ‘<’ is mapped to the previous one because the right and left direction is usually mapped to ‘next’ and ‘previous’.

The design of digit shapes is complicated because writing styles are different according to nationalities, ages and genders of users. Therefore, we surveyed 120 people in our company on their writing styles. Table 1 presents the most popular digit shapes except 6 whose ending part becomes elongated for discrimination from 0.

4.2. Data Collection

In order to evaluate the proposed system, we collected data from 100 writers. Because the cell phone is targeted for young generation, all the writers were of 20’s and 30’s and did not have any experience of using it. The phone was attached to a PC by serial port during data collection. Acceleration signals were generated from the phone and then transmitted and saved in the PC.

Figure 8 shows the proper hand posture and the activity in data collection. A user draws gestures of 14 classes (1-9, O, X, <, >, M) by three times while looking at their labels and representative shapes shown on PC screen. They were asked to hold the phone with its screen facing up about 60 degree from the earth plane and write characters on imaginary vertical plane in front of them. Because the gesture activation button is located in the right side of the phone, the hand stays the most comfortably at the posture.

4.3. Evaluation Result

The recognition performance is measured with 11 gestures (1-9 and O, X) that are recognized at the same time in the idle state. The other states such as MP3

$$\text{Fig. 8. Data collection activity}$$
played or message received have only 2 classes and 1
class to recognize respectively. Therefore, the idle state
has the most difficult recognition task.

In order to measure the user independent recognition
performance, 100 users’ database were divided into four
folds without any common data. Then, four-folds
generalization test was performed and the four
recognition rates were averaged.

Table 3 shows the confusion table of BN based
recognizer. The row and the column denotes data classes
and the recognized class. The average recognition rate is
96.3%. The table shows that almost 30% of errors come
from the pair of O and 6 (35 among 114 errors).

<table>
<thead>
<tr>
<th>Recognized as</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>258</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>16</td>
<td>(8)</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>289</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>282</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>288</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>289</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>289</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>19</td>
<td>(9)</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>246</td>
<td>(250)</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>255</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>255</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>3</td>
<td>0</td>
<td>249</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>291</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Confusion table for the gesture recognizer:
( ) in the cells denote results by SVM

The SVM discriminator for 0 and 6 was trained with the
feature of 10 resampled points as explained in Eq.
(14). Its four-fold generalization result was 97.6%.

By combining the BN classifier and the SVM pairwise
discriminator, the recognition performance is further
enhanced (Table 3, digits in ()). The combination scheme
is as follows: First, a data is classified by the BN
recognizer. If the recognition candidate with top score
belongs to either O or 6, SVM pairwise discriminator is
called. Otherwise, the recognition candidate becomes
the final recognition result. Table 3 shows that about half of
errors in O and 6 are resolved so that the overall
recognition rate is 96.9%. Even though SVM can classify
O and 6 with the accuracy of 97.6%, the errors were
reduced only by the half in the final result because the
errors by SVM and BN are largely overlapped.

5. Conclusions

In the paper, we presented a new kind of interaction
method based on a hand motion as intuitive and
convenient input methods. People can make speed
dialing, navigate songs, delete messages, and generate
gesture sounds by drawing their gesture shapes in the air.
The interaction method has the advantage that users do
not need to pay attention to tiny keypads. Our algorithms
were employed in the world-first motion-understanding
phone, Samsung SCH-S310 and E760.

The gesture recognition algorithm models basic
components and their relationships of acceleration
signals. Acceleration signals are normalized by removing
gravity components and writing speed variations. Then,
the signals are divided into primitives at feature points.
The primitives and their relationships are modeled with
conditional Gaussian distributions. The robustness of its
recognition capability is further enhanced by discriminating confusing pairs based on SVMs.

We evaluated the performance of the algorithms with
100 users who did not have any experience of using the
phone. For 11 gestures (digits 1-9, O, X), the average
recognition rate was 96.3% with Bayesian networks.
About 30% of the recognition errors came from the pair of
(O, 6). With pairwise discriminator, the recognition
rate was further enhanced into 96.9%.

Our next research is targeted for making motion-based
interaction as one of basic interaction means in mobile
devices. For the purpose, continuous gestures need to be
recognized for inputting consecutive digits, English
words or Korean characters, which will enable people to
input short messages by drawing them in the air.

6. REFERENCES

1. E.-S. Choi, et. al., “Beatbox Music Phone: Gesture
Interactive Cell phone using Tri-axis Accelerometer,” IEEE
Int. Conference on Industrial Technology, 2005
Cell phone,” Proc. 7th Int Conf on Human Computer
Interaction with Mobile Devices and Services, 2005
3. Article, “Digital Appliance, Controlled by Movement,”
Nikkei Electronics April 11, 2005
Interfaces (Tech Note),” UIST, 1996, pp. 167-168
“ReachMedia: On-the-move interaction with everyday
objects,” ISWC, Osaka, Japan, 2005
Authentication Scheme for Untrusted Public Terminals,”
UIST, Santa Fe, USA, 2004
Users to Easily Establish Connections between Smart
Artefacts,” Ubicomp, Atlanta, USA, 2001, pp 116-122
device in 3-D space with inertial sensors,” IWFHR, Tokyo,
2004
wearable computers,” JSWC, USA, 2003, pp. 26–34.
10. J. Mäntyjärvi, et. al, “Enabling fast and effortless
customisation in accelerometer based gesture interaction,”
MUM, College Park, Maryland, 2004 pp. 25-31.
11. Samsung SCH-S310 product web site (in Korean),
PPID=SCH-S310&pages=1&real=feature&srar=
12. Samsung GSM E750 product manual,
http://www.samsung.com/uk/products/mobilephones/mobil
ephones/sgh_e760saxeu.asp (in English)
14. F. Jensen, An Introduction to Bayesian Networks, Springer,
New York, 1996
15. S.-J. Cho et. al., “Bayesian network modeling of strokes
and their relationships for on-line handwriting
recognition,” Pattern Recognition, vol. 37, no. 2, pp. 253–
Characters for On-line Handwritten Recognition,” proc. 7th
ICDAR, 2003, pp 207-211
17. N. Cristianini et. al., Introduction to Support vector
machines , Cambridge University Press, 2000