Learning People's Perception of Messiness for Hand-drawn Sketches

by

Banuçiçek Gürcüoğlu

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Computer Engineering

Koç University

August 26, 2014

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Banuçiçek Gürcüoğlu

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Assist. Prof. T. Metin Sezgin, Koç University

_____

Assoc. Prof. Engin Erzin, Koç University

_____

Assist. Prof. Albert Ali Salah, Boğaziçi University

Date: _____

to my family

ABSTRACT

The increasing availability of pen-based tablets and pen-based interfaces resulted in several lines of work that attempt to build computational tools for processing sketches. These include computational methods for classifying sketches and methods for beautifying rough drawings. In this thesis, we pose discrimination of clean sketches from messy ones as a new problem, and propose a computational framework for learning messiness from examples. Our system does not use domain, object or context specific knowledge. Our approach is inspired by human perception and can assess the messiness of hand-drawn sketches using features extracted from relative orientation, length and distance properties of primitives. In order to evaluate our method, we collected a sketch dataset that consists of 1535 drawings of 32 different symbols, which were also coded with subjective messiness information obtained from human raters. Our evaluation shows that we can discriminate messy and clean drawings with up to 90% accuracy when the problem is treated as a 2-class classification problem. We report detailed analysis on what makes this problem hard, and how the system accuracy relates to the inter-rater agreement on the messiness ratings of sketches.

ÖZETÇE

Kalemli tablet bilgisayar ve kalemli bilgisayar arayüzlerinin yaygınlaşması, birkaç çeşit otomatik çizim işleme çalışmalarına yol açtı. Bu çalışmalardan bazıları çizim tanıma ve taslak çizimleri güzelleştirme yöntemlerini araştırır. Bu tezde, özen gösterilerek çizilmiş görünen çizimleri özensiz çizimlerden ayırmayı, çözülmesi gereken yeni bir problem olarak ortaya koyup, çizimlerin ne kadar özenli göründüğünü örneklerden otomatik olarak öğrenen bir sistem sunuyoruz. Sistemimiz alan, nesne ya da bağlam bilgisi kullanmamaktadır. Çözüm yolumuz, insan algısından esinlenerek, çizimlerin basit parçalarınının yön, uzunluk ve uzaklık bilgilerini kulllanarak ne kadar özenli çizildiklerini ölçebilmektedir. Yöntemimizin başarısını ölçebilmek için, aynı zamanda insanlar tarafından ne kadar özenli çizildikleri değerlendirilmiş, 32 farklı sembolün 1535 çiziminden oluşan bir çizim veri kümesi derledik. Değerlendirmemiz gösteriyor ki, problem iki sınıflı sınıflandırma problemi olarak ele alındığında, sistemimiz %90'a varan bir başarıyla özenli çizimleri özensiz olanlardan ayırabilmektedir. Ayrıca problemin zor olmasına sebep olan etkenler ve sistemin başarısı ile çizimlerin ne kadar özenli göründükleri hakkında insanların ne kadar mutabık olduğu arasındaki bağlantısı üzerine analizlerimizi de bildiriyoruz.

ACKNOWLEDGMENTS

There are many people I would like to thank for their input and support during my studies at Koç University.

First and foremost, I would like to thank my thesis advisor T. Metin Sezgin, for his support and guidance during my studies as he was always available and patient with my learning process.

I would also like to thank my thesis committee members, Engin Erzin and Albert Ali Salah for their valuable input.

I am most grateful to my parents Orhan Gürcüoğlu, Nevin Gürcüoğlu, and my sister Bengisu Gürcüoğlu, for their endless emotional support.

I would also like to express my gratitude to Levent Yaldız for his caring patience throughout the course of this thesis and especially for his support during the sleepless nights and psychologically challenging times.

I express my warm thanks to all of my friends. I thank my cousin, Görsev Argın for the enormous help she provided. I would also like to thank Neşe Alyüz, Deniz Turgay, Özem Kalay, Çağla Çığ , Burak Özen and all the people in my research group and colleagues at Koç University.

I would also like to thank Tijen Güreşen, for sharing her home with me and saving me tens of hours during the last few months.

I would also like to thank TÜBİTAK, for the financial support under the TÜBİTAK-BİDEB 2210 - National Scholarship Programme for MSc Students, and people who participated in the data collection and the psychophysical experiment phase of this study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Motivation and Problem Statement

Freehand sketching is a natural means of communicating ideas. Sketches allow us to express complex ideas succinctly in ways that are not possible with words. These properties of sketching, coupled with the widespread availability of pen-based computers, have made sketching the subject of extensive research in the human computer interaction community. In particular, there has been a surge of interest in computational methods for processing sketches. However, the efforts so far mainly focused on recognition and beautification of sketches. In this thesis, we set forth the problem of assessing the messiness of hand-drawn symbols as yet another computational problem that deserves the attention. The practical utility of sketch recognition and beautification are rather clear. However, the potential uses of assessing messiness may not be as immediately obvious. Hence, it is worth enumerating a few potential use cases. For example, consider a tutoring system that tries to improve peoples drawing abilities. Our system would be useful for reporting the observed improvements of the sketches in terms of their neatness. Messiness can also be used for organizing/visualizing/indexing large databases of sketches by finding clean prototypes. Messiness is an external source of knowledge to obtain confidence in sketch recognition results, as we report the strong correlation between the messiness scores and symbol recognition accuracies.

In this thesis, we combine the notion of aesthetics with hand-drawn sketches that consists of geometrical sub-parts and aim to discriminate messy sketches from neat

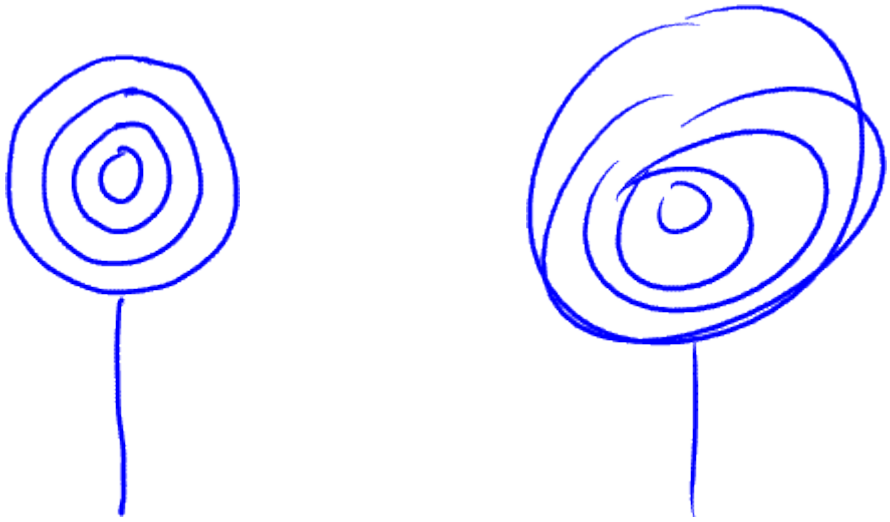


Figure 1.1: Samples for neat and messy drawings

ones, such as seen in Figure 1.1, based on the people's perception.

1.2 The computational approach

We learn computational models for assessing messiness from examples. Hence we require a dataset of symbols with ratings of messiness. We compiled such a database by collecting 1535 sketches, on a tablet PC using a stylus, from eight users of various backgrounds and ages under different condition, and conducted an extensive controlled psychophysical experiment for ranking and rating each sketch by four individuals. Our sketch dataset consists of 32 different familiar symbols, consisting of combinations of lines and ellipses. Symbols from each symbol class were ranked and rated based on how messy they appear to subjects in a single session through a table scaling experiment [Rogowitz et al., 1998]. Later, this table scaling approach was validated by the statistical difference between human and random orderings using the variance of ranks [Adali et al., 2007] metric, we computed messiness scores for each sketch by taking the means of their ranks by four subjects. Our analysis on the subjective messiness information showed high consensus between people on neat and

messy ends of the messiness scale, consensus dropped towards the middle of the scale.

Once the dataset is obtained and the subparts of the sketches annotated as lines and ellipses via a semi-automated method, we extracted features. Initially we treated the messiness prediction problem as a 2-class classification problem with classes being messy and neat. We tested our messiness features under different conditions and see how accurately they learn the people's perception of messiness. Those conditions varied in terms of subject agreements on messiness, how apart the messy and neat classes on the messiness scale, size of the training set and test set consisting of unseen symbols.

1.3 Related Work

We study computational aesthetics of hand-drawn sketches, based on human perception. Therefore, we will discuss related work on computational aesthetics, sketch beautification systems and also discuss the studies that use different methods for measuring human perception.

1.3.1 Computational aesthetics

Visual aesthetic judgment has been an area of interest in many context. Some of these contexts are photographic images, which is rather popular, color, handwriting and graph drawing. In a study by Datta et el., automatic aesthetics inference of pictures form their visual content is treated as a machine learning problem [Datta et al., 2006].

In another study, Li et al. automatically assess the quality of photos, using the technical characteristics of the photo and the specific features related to faces [Li et al., 2010]. Wu predicts aesthetic adjectives of images using SVM-based classifiers [Wu et al., 2010], whereas, Wong predicts the aesthetics classes, obtained from aesthetics scores of images on Photo.net [Wong and Low, 2009].

O'Donovan et al. learn preferences for color aesthetics, using collaborative filtering approach and uses a large dataset of rated color themes/palettes [ODonovan et al., 2014].

Study by Arsdale investigates the *neatness* of the student works via spatial and

temporal features and conclude that the neatness, organization of a students answers on paper can explain about 40% of the variance in students' performance on the questions [Arsdale and Stahovic, 2012].

There are some work done on layout aesthetics of graph drawing. One of them stands out, as they focus on the creation of the graphs by reporting on an experiment where they ask the participants to lay out adjacency lists graphs "nicely" [Purchase et al., 2012].

1.3.2 Geometrical constraints

Even though there is no previous work on assessing the messiness of a sketch, there have been number of studies regarding the beautification of sketches. These beautification systems *neaten* drawings by ensuring some geometrical constraints and/or smoothing the strokes and none of these studies perceptually assess how successful is the resulting beautified version of the sketches. Instead, are evaluated by the improvement they provide to the systems they are part of, e.g. by reducing the completion time of a designing task using the system in question [Igarashi et al., 1997], or often not evaluated at all.

Existing beautification systems support different lists of geometrical constraints. In a study by Igarashi et al., geometric designs are interactively beautified by satisfying a set of geometric constraints, such as connection, parallelism, perpendicularity, alignment, congruence, symmetry and interval equality [Igarashi et al., 1997]. Cheema et al. recognize diagrams consisting of line segments and circles, and infer geometrical constraints between these diagram components [Cheema et al., 2012]. Murugappan et al. use an interface that suggests correction of the input sketch to the users. These suggestions are the versions of the input sketch where sets of geometrical constraints are satisfied [Murugappan et al., 2009]. In another study, a statistical approach called Relative Shape Histogram is used to detect the geometric constraints intended by the sketcher and a priority-based approach is presented for choosing which geometric constraints to satisfy on graph designs [Pu and Ramani, 2007]. The system

in [Paulson and Hammond, 2008] beautifies sketches by returning the geometrically standardized versions of the shape objects it recognized. Apart from these beautification systems, Veselova et al. treat the geometrical features as perceptual properties of sketches and learns descriptions of hand-drawn symbols from a single example using these properties [Veselova and Davis, 2004].

1.3.3 Measuring the human perception

While trying to come up with a experimental design for measuring the human perception of sketch messiness, we conducted an extensive literature review. Common issue encountered was the trade off between number of participants and viability of a controlled-user study. If high number of participants was aimed, then the experiments were conducted in an uncontrolled environment using crowd-sourcing.

Vatavu et al. automatically estimate the perceived difficulty of single-stroke pen gestures [Vatavu et al., 2011]. Their method for measuring the difficulty is a combination of ranking and 5-point Likert scale rating similar to our method for measuring sketch messiness. Rogowitz et al. compare two psychophysical scaling experiments, table scaling, which we ended up adopting, and computer scaling, in the context of image similarity [Rogowitz et al., 1998]. Mantiuk et al. compare 4 different subjective image quality experiments [Mantiuk et al., 2012]. These experiments are listed as *single stimulus*, *double stimulus*, *forced choice* and *similarity judgments*. Clarke tests the reliability of a free sorting experiment via pairs of pairs experiment in the context of visual similarity between pairs of textures [Clarke et al., 2012].

Some of the studies discussed under the topic of computational aesthetics also shed light on how to measure human perception. Datta use a peer-rated online photo sharing Website (Photo.net) as data source for their challenge of automatically inferring aesthetic quality of pictures using their visual content [Datta et al., 2006]. The also state that they wished to collect the data under controlled setup, but resource constraints prevented them from doing so. In the study by Wu, the images are gathered by using English language queries for 6 adjectives on image sharing website, flickr

and train SVM-based classifiers to predict these aesthetic adjectives [Wu et al., 2010]. Wong et al., too, make use of the Internet and get images on Photo.net where each image has an aesthetics score from one to seven and use the top 10% and bottom 10% as their dataset [Wong and Low, 2009].

Chapter 2

DATA COLLECTION

As our aim is to assess the messiness of sketches, we need a database of hand-drawn sketches. Furthermore, each sketch in this database should have an assessment of its messiness to serve as ground truth ratings. We build such a database in two steps:

1. Compile a large database of sketches

2. Obtain reliable ratings of messiness for each symbol.

We, now, describe these two steps in detail.

2.1 Sketch Data Collection

Obtaining generalizable features for assessing the messiness has been one of the main objectives of this study. Therefore we adopted a set of diverse design choices to maximize data diversity. In particular, we took the following measures:

- We tried to create a variety of conditions to promote messy as well as clean drawings.

- We used a large number of symbols.

- We chose a set of symbols that contain diverse set of constraints within its constituent primitives, i.e. geometrical subparts such as line and ellipse.

Reasons for messiness There may be various reasons for a sketch to end up as a messy one. These reasons may reflect differently on the features level and it is not possible to foresee. Therefore, we aimed to keep the list of messiness reasons as long as possible. Some of these reasons are scientifically reported that they affect the quality of a drawing, and some of them are based on our intuitions and experience. The resulting list of reasons for messiness covered in this study is as follows:

1. Rush while sketching
2. Boredom while sketching
3. Negligence while sketching
4. Age of the sketcher as it affects the developmental skills [Kim et al., 2013]
5. Level of experience with tablet
6. Level of experience with drawing or level of drawing talent

We aim to simulate the effects of the items in the above list by varying the sketching modes during the data collection, and the age and experience level of the participants. In Section 2.3.1, we report that these effects are visible on the subjective messiness scores as we anticipated, once we collect the subjective messiness dataset. Even if we do not have enough data to prove conclusively (as it is beyond the scope of this study) that the items in the above list have certain effect on drawing, the correlations between messiness scores and the variables of the data collection are quite strong. Please see Figures 2.14 and 2.15 for the correlation coefficients.

Primitive Types and Geometric Constraints Primitives of a sketch are the smallest meaningful geometrical subparts of the sketch. Since we will extract our features for messiness prediction at a primitive level, we need the primitives of sketches in our dataset to be annotated as lines and ellipses. Annotation of these primitives are

called fragmentation of sketches and it is an ongoing research area. State-of-the-art fragmentation methods still have some limitations. In this study, we preferred to get pass the issues of fragmentation, and extract the features free of annotation errors. As the size of the dataset makes the manual annotation not feasible, we decided to limit the types of primitives in our dataset to lines and ellipses, so that a semi-automated primitive annotation, which will be explained in detail in Section 2.1.2 can be utilized.

Even though we limited the number of primitive types due to fragmentation issues, we aimed to maximize the number of geometrical constraint types, that can occur between these primitive, covered by our sketch dataset. Geometric constraints supported by CAD systems such as SOLIDWORKS and the list of geometric constrains that are covered in beautification system developed by [Cheema et al., 2012] [Igarashi et al., 1997], [Murugappan et al., 2009], [Pu and Ramani, 2007] and [Veselova and Davis, 2004] have been reference while we were making the lists of geometric constraints that will be covered by our sketch dataset. These constraints can be categorized into five groups according to the types of the primitives they occur between:

1. *point to point*: horizontal, vertical, coincident

2. *point to line*: midpoint, collinear

3. *line to line*: parallel, perpendicular, same length, equal, collinear

4. *line to arc*: tangent, collinear

5. *arc to arc*: co-centric, same length radius, tangent, co-radial, horizontal, vertical

Aside from these geometrical constraints, symmetry is also a factor when people assess the messiness.

Our aim was to collect a sketch dataset that contains variety of these constraints between the primitive types, line and ellipse.

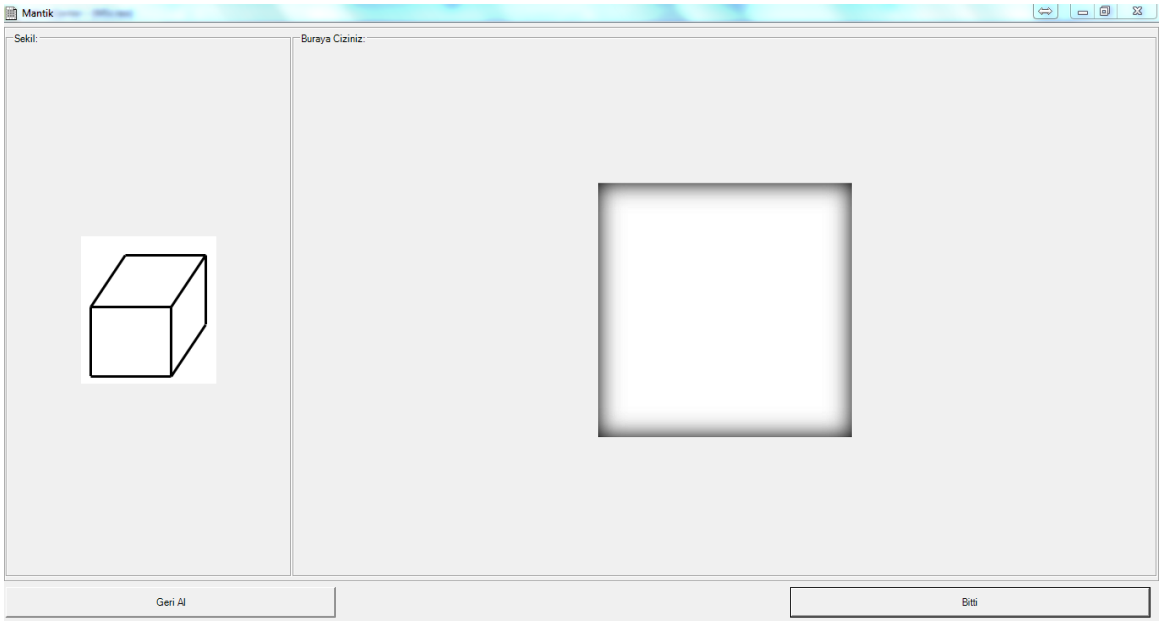


Figure 2.1: The interface of the software used to collect the sketches.

Size of the dataset: Generalizability issue required the number of sketch samples from individual symbols and individual sketchers to be high as well as the number of different symbols. Resulting dataset was aimed to be suitable for various kinds of analysis such as symbol-wise messiness and sketcher-wise messiness.

2.1.1 Conducted data collection

Sketches are collected using a Tablet PC (Lenovo ThinkPad X200) via a sketch collection software developed by IUI Laboratory at Koç University. The interface of this software can bee seen in Figure 2.1. Participants were asked to draw the symbol they see in the box on the left, on the canvas which is the box on the right. Tablet PC was placed on a table and drawings were done using the stylus. They had the undo option, which deletes the last drawn stroke when the button on the bottom left with the label 'Geri Al' is clicked. When they were satisfied with their drawing, in order to proceed to the next symbol, they clicked the *done* button on the bottom right, labeled 'Bitti'.

Symbols As it can be seen in Figure 2.2, the symbols that are collected are in the form of one of the following three categories:

1. *all lines*: arrow, box, braid, building, envelope, forward, house, ladder, lamp, lightning, pause, rug, snowflake, star, table, tent, tiltedladder, tree, wall

2. *all ellipses*: olympicsymbol

3. *combination of lines and ellipses*: ant, bow, candy, clock, fishbone, leaf, racket, stickfigure, sun, trafficlight, train, wheel

Phases In order to cover messiness reasons such as boredom or rush in our data collection, we altered the sketching mode of the participants during the sketch data collection. The data collection consisted of three different modes of sketching, that we call phases. In each phase, 32 symbols were presented twice in random order to the participants. The phases, in the order they appeared in the data collection, are as follows:

1. **Normal phase**, was for the participants to get comfortable with the symbols they were asked to be sketched. There were no limitations on time or outcome of their sketching.

2. **Careful phase**, was the phase where participants were asked to "take their time and do their best to make the sketches as neat as possible".

3. **Fast phase**, was the phase where participants asked to conclude the data collection as fast as possible, without over thinking about the outcome sketch. Aural encouragement was also given by the conductor of the data collection in order to create feeling of rush.

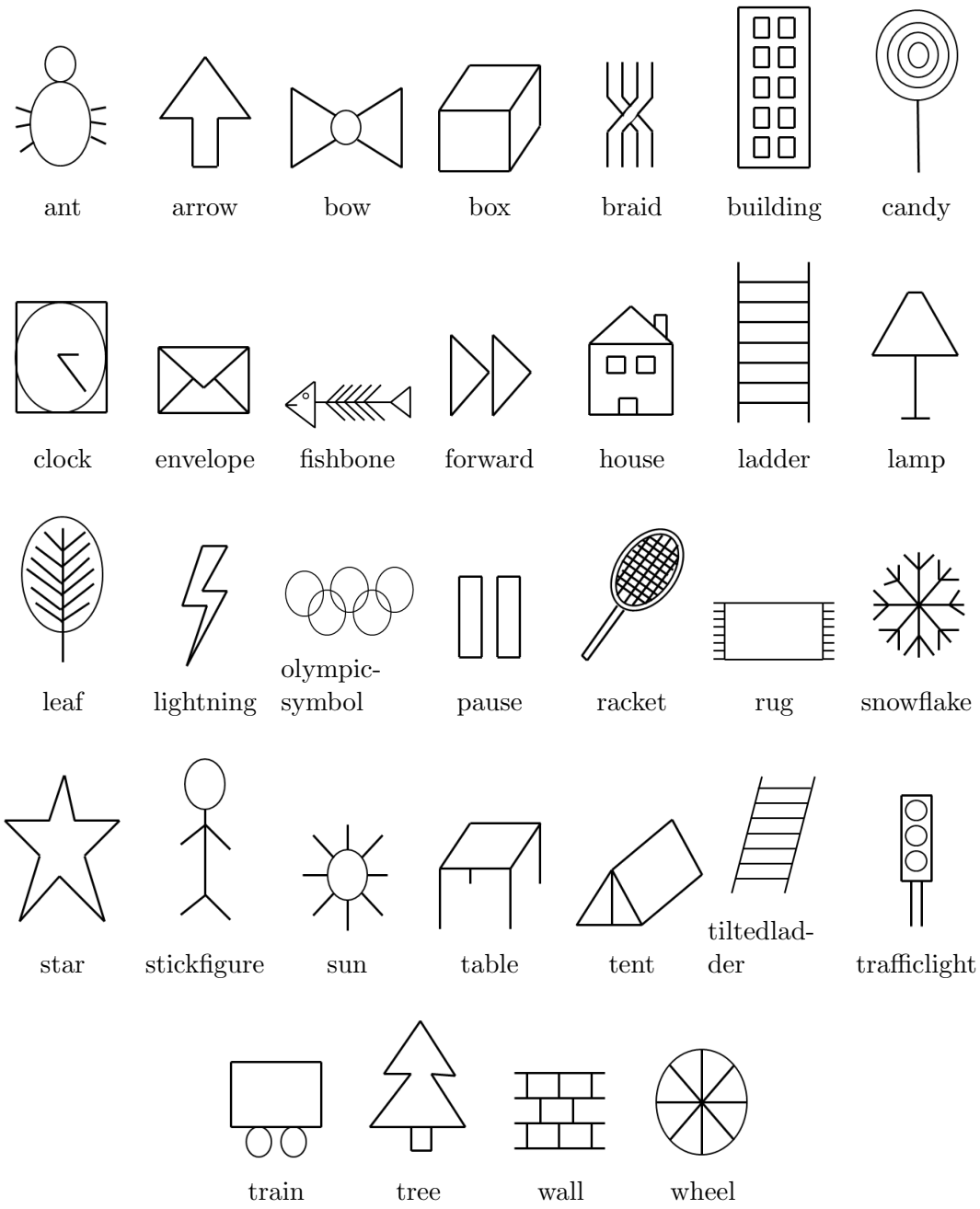


Figure 2.2: Symbols in the sketch dataset

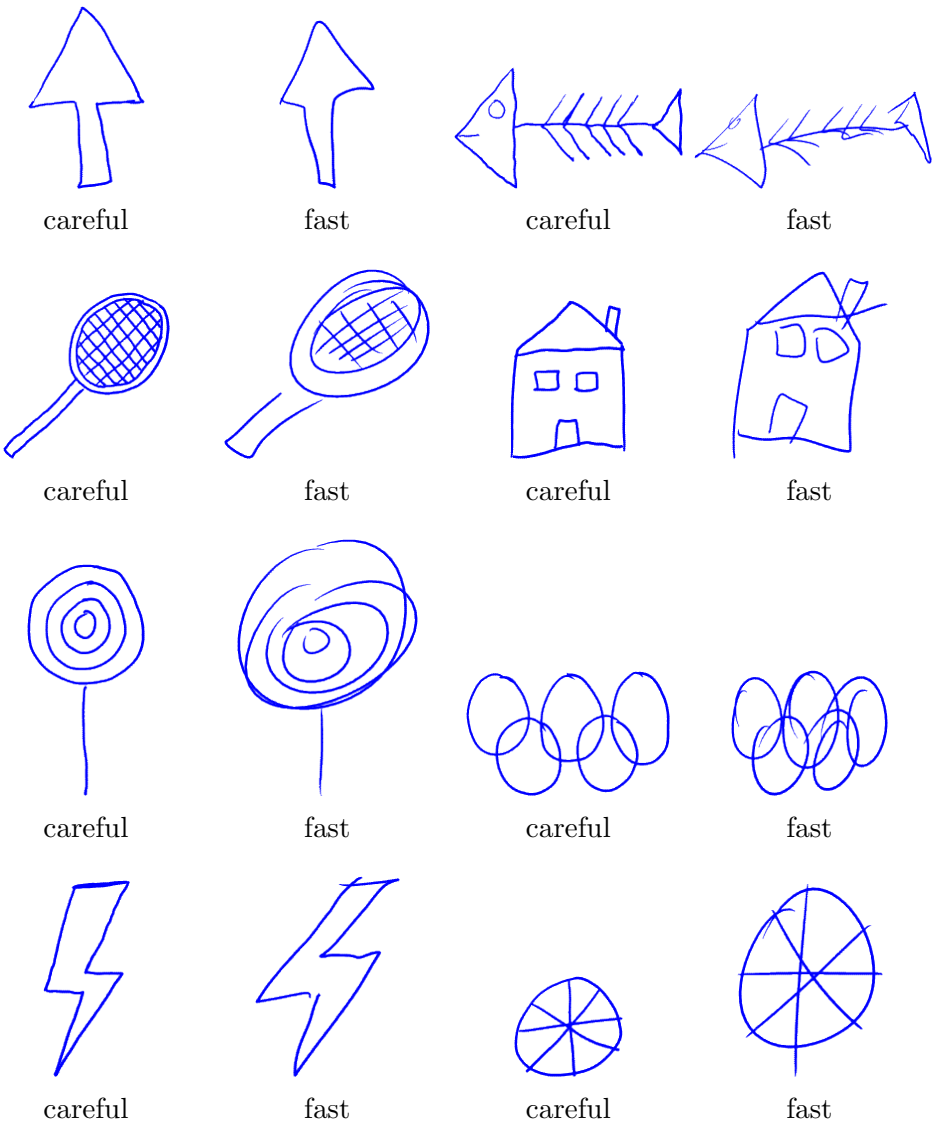


Figure 2.3: Careful versus Fast Phase. Sketches of the same symbol by the same sketcher from careful and fast phases, respectively.

Participants Sketches are collected from eight people (five female - three male), whose year of births differ between 1964 and 2002. One of the participants is a designer who has experience with drawing on a tablet using stylus.

As mentioned before, the correlation between the age of the sketcher and the messiness score, as well as the correlation between the sketching mode and the messiness score, are further discussed in Section 2.3.1, as the subjective messiness dataset is obtained at that stage.

Resulting Dataset consists of 32 symbols x 3 phases x 2 repetitions x 8 sketchers = 1535 sketches in total. A sketch from the symbol, stickfigure, was not saved during the data collection.

2.1.2 Primitive Annotation of the Sketches

Primitive annotation of the sketches are done via a semi-automated method using hard-coded thresholds. The algorithm first finds the corners by Douglas Peucker corner finding algorithm, which is set empirically in order to minimize the number of missed corners as well as the number of points belonging to the lines but recognized as corners by corner finding algorithm because the line is drawn roughly. Then every stroke is divided from the corners, found by Douglas Peucker, into primitives. Then an ellipse is fit to each primitive and seen if various measurements are between certain thresholds that are set by us according to our dataset. If the primitive follows the rules of being an ellipse, then marked as ellipse, otherwise, as line. In Figure 2.4, primitives of some symbols are shown with a color code, each color representing an individual primitive.

2.2 Subjective Messiness Measurement Experiment

Methods for measuring subjects' perception on things are collectively called *subjective methods* [Mantiuk et al., 2012]. As there is no previous work done on sketch messiness, to our knowledge, we mainly focused on subjective methods for image

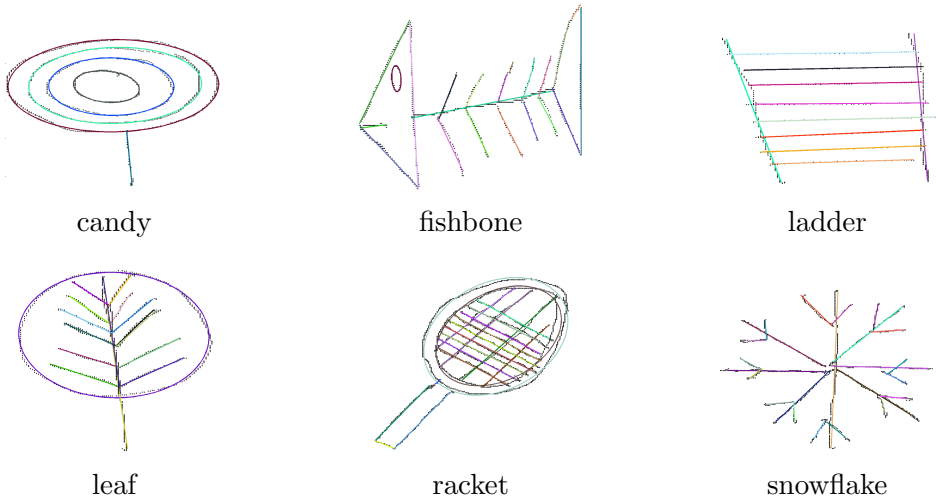


Figure 2.4: Primitive annotation of some symbols. Each primitive annotated by a different color.

quality assessment. Different types of subjective methods are proven to be suitable for different purposes on assessing people's opinions. In our case, we had 32 different symbols which has 48 sketches each, giving 1535 sketches in total. Our aim is to collect the information of how messy each sketch in our dataset is in order to use them as reference for our sketch messiness prediction system. The size of the sketch dataset required us to divide the dataset and conduct the experiment with several subsets of the dataset. Our preliminary approach was to ask subjects to rate the shown sketch using a 7-point Likert scale. The sketches to be shown were selected randomly from the whole dataset. The results were highly unreliable, as it generally is with rating systems [Jin et al., 2003]. This preliminary experiment demonstrated how hard it was for subjects to form a messiness scale on their mind when messiness of different symbols were in question. Our discussions with the participants revealed that this was due to the fact that there may be several aspects of a sketch that causes it to be messy. These aspects can be summed up as follows:

1. smoothness of the lines

2. the sketch's resemblance to the symbol that the rater thinks was intended to be

drawn

3. whether the geometrical constraints that are intended between some primitives of the sketch was indeed accomplished

Participants told they were confused by the number of aspects they needed to consider and it was hard for them to form a judgment combining these aspects. As a result, people could not just look at a single drawing and rate its messiness. They required some kind of reference.

In the light of the preliminary experiment, we decided to collect messiness information of a sketch relative to the other sketches of the same symbol. Our ultimate goal was to obtain the overall ranking of all the sketches in its own symbol class. For this purpose, we also utilized a rating with a 5-point Likert scale, in order to ease the ranking process of 48 items of the same symbol as suggested by [Vatavu et al., 2011], and also to have the labels in case they are necessary. In our experience, dividing the sketches into categories helped the subject to handle the ranking task and gave them a starting point.

2.2.1 Conducted Experiment

Table Scaling Our aim was to get an ordering of all sketches of a symbol at once and we preferred for the participants of the experiment to be able to see all the sketches to be ordered at once. A computer interface would be limiting for this purpose due to the restricted size of any possible display. Instead, we chose table scaling [Rogowitz et al., 1998] as our medium, i.e. we printed out all the drawings, and asked the subjects to order these printouts on a table. As table scaling frees the subjects from interface related restrictions, it poses the problem of digitizing the input.

Setting In our case, we planned for two subjects to order each symbol at the same time and four symbols in average during one session. In order to avoid making the

subjects wait for their inputs to be recorded and the table to be cleared for the next symbol ordering, we decided to divide a large enough table into four parts. The experiment was conducted in a meeting room (ENG 208) at Koç University. The experiment setting can be seen in Figure 2.5. Each side of the table belongs to one subject and each side can accommodate two orderings at a time, one to the front of the red dots and one to the back.

For the experiment, we printed out two sets of the whole sketch dataset on papers with white background. Each print out had a QR code of the name of the sketch file on the bottom right of the print out, in order to be recorded via a barcode reader. A sample sketch print out, as used in the experiment, can be seen in Figure 2.6 in its actual scale.

When an ordering was finished on one half of the table, the experiment conductor recorded the orderings via reading the QR codes of the each sketch in order and cleared the table for the next ordering, while the participants ordered the next symbol on the other half of the table. Barcode reader application called QuickMark, which allowed fast and exportable scanning, is used in order to scan the QR codes of the sketches. By utilizing QR codes, we reduced the required time for recording 48 sketches in order excessively and eliminated the human error factor during this process.

Participants We recruited 28 students (17 male, 11 female) from Koç University for the experiment. Four of the subjects participated in two sessions. Dates of birth of the participants differ between 1981 and 1996.

Motivation for Participants Participants competed for a money prize of 8 Turkish Liras for ordering of one symbol in order to prevent negligence. They were told that there was a true ordering of each symbol and the participant who gets closer to this true ordering gets the money prize. The actual mechanism behind the prizing was that, each participant had to complete even number of orderings and were told that they won half of the orderings, where in fact they both receive half of the money for each ordering.

Figure 2.5: subjective messiness measurement experiment setting. Each side of the table belongs to the corresponding subject and each side is divided into two parts designated by the red dots.

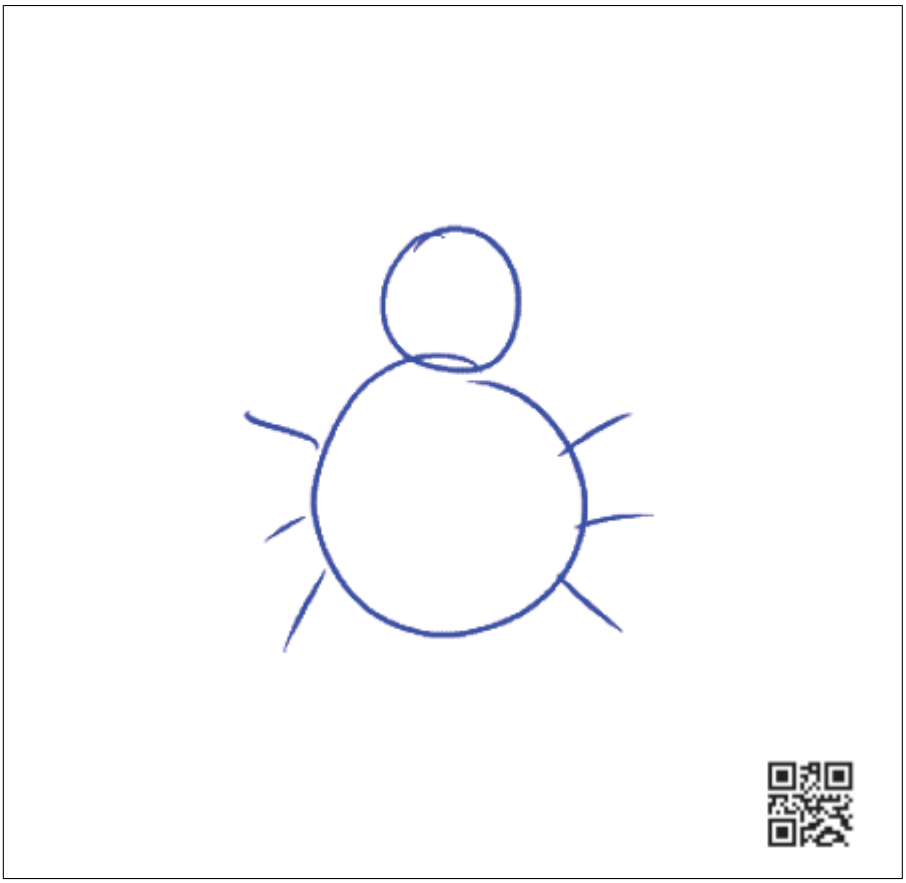


Figure 2.6: QR code representing the file names of the sketches

A session The experiment was explained to the participants in a 15 minutes demonstration of the competition between them using a random test symbol. After it is made sure that both participants understood the rules and what was expected of them, the actual experiment has started. Given 10 minutes and an envelope containing all sketches of a symbol in random order, they were asked to

1. Lay down 48 sketches of a symbol on the table.

2. Make sure that there are 48 sketches, in case some papers stuck together.

3. Divide the sketches into the 5 categories on the table: very messy, messy, neither messy nor neat, neat, very neat. Number of sketches in each category is not restricted in any way.

4. Order the sketches according to how messy they are in each category so that sketches end up in a linear ordering from very messy to very neat. This, eventually gives each sketch a ranking within its own symbol class.

Figure 2.7 demonstrates the setup of the table for the case of both subjects ordering the sketches of a symbol.

Time restriction Time restriction of ten minutes for each ordering was designed for the purpose of preventing subjects from over thinking, This ensured that they rank the sketches according to how they initially perceived the messiness. 10 minutes was decided on in the light of the preliminary phase of the user studies. There was no case where a subject ran out of time before forming the initial ordering during the actual experiment. Some of the participants fully made use of the time given (e.g. switching sketches until the last second), whereas some participants stopped ordering before their time was up.

Scale of the experiment In each session two participants ordered four symbols in approximately an hour. There were 16 sessions in total, where each symbols were ordered four times. Total of 512 Turkish Liras was distributed during the experiment.

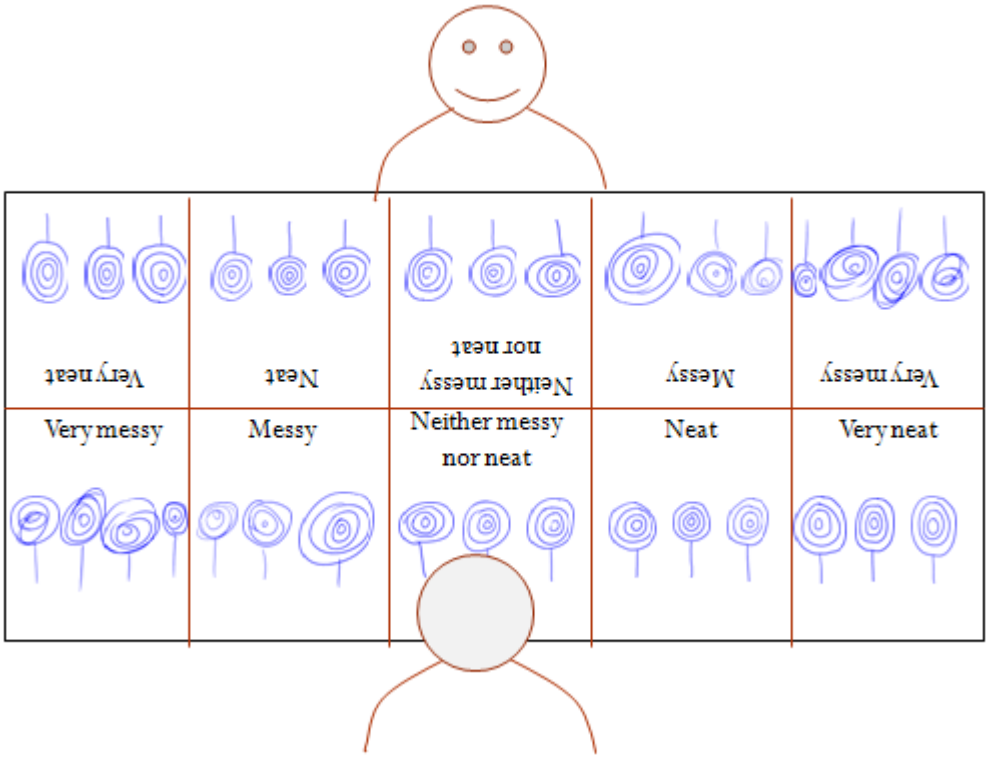


Figure 2.7: subjective messiness measurement experiment table setup

Resulting Dataset The resulting subjective messiness dataset is, for each sketch, obtained from 4 individuals:

1. *a label*: being one of the following; very messy, messy, neither messy nor neat, neat, very neat

2. *a ranking*: giving the ranking (1-48) of the sketch among its own symbol class

2.2.2 Reliability of the Subjective Messiness Dataset

Reliability of a subjective methods is a common concern and hard to achieve. In order to validate the subjective messiness data we have collected, we have done various analysis on how the subjects agree. As we got the sketches both rated and ranked, we analyzed the agreement of subjects on both.

Rating

It is very challenging to achieve high inter-rater and intra-rater agreement for subjective rating. Subject may change the rating scheme they use through out the study. And the subjective nature of the process, most probably, cause low inter-rater agreement. [Jin et al., 2003] states that, even two users with very similar preferences may have very different rating schemes. We report the agreements on the labels of the sketches without any post-processing in order to normalize the rating schemes of the subjects, as we used rating mainly as an aid for the ranking process. Figure 2.8 shows the number of unique rates/labels for all the sketches in the dataset, after the sketches are ordered from neat to messy according to the mean of their ranks by all subjects.

Ranking

As a metric for assessing the reliability of the ranking we used what we call *index difference* which corresponds to *variance of ranks* in [Adali et al., 2007].
Let $\text{rank}(r_n, s)$ be the rank of sketch s by the n^{th} ranker of the symbol that s belongs to, index difference is calculated as $\text{d}(\text{rank}(r_i, s), \text{rank}(r_j, s))$ for the i^{th} and j^{th} rankers

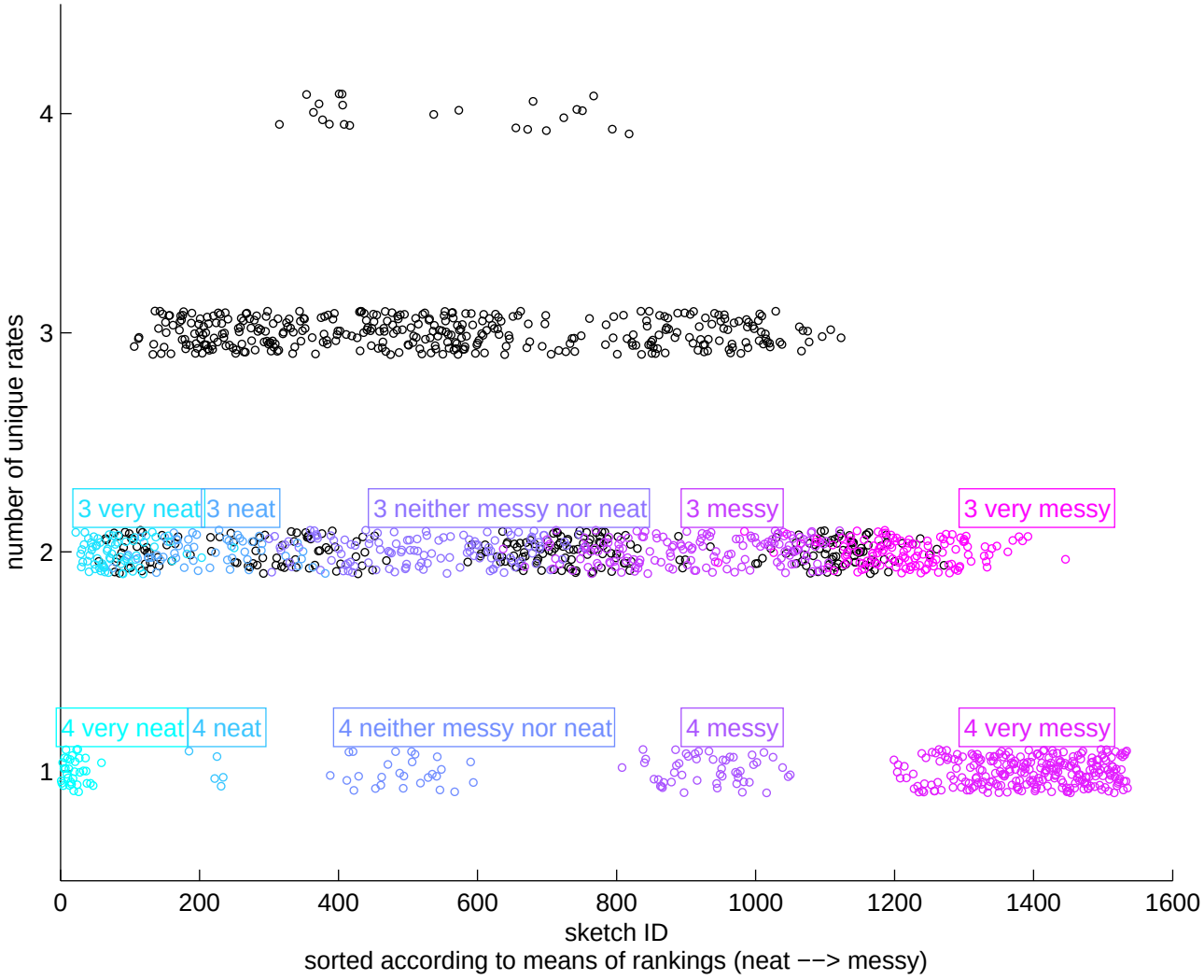


Figure 2.8: Number of unique rates for each sketch. All the sketches are ordered from 1 to 1535 according to their messiness score, 1 being the neatest and 1535 being the messiest. Color codes show the sketches that three or four subjects agree on the rates of.

of s. For example, one of the subjects ranked the sketch 7^{th} in its symbol class, another subject ranked the same sketch as 10^{th}, then the index difference of the sketch for these two subjects is $10 - 7 = 3$.

Figure 2.9 shows the average of the index differences for all sketches of individual symbols for the all possible subject pairs for that symbol. Top five rows shows what would this metric yield if the ordering was a random one, by comparing random ordering to each subject of a symbol and also comparing a random ordering with another random ordering. Different random orderings are produced for each symbol as if they are the fifth and the sixth subjects.

Average of the index differences for all possible subject pairs of all the sketches is 6.17 out of 48. It is seen that there is a peek in index differences for the 3^{rd} subject that ranked forward, and 2^{nd} subject that ranked tent, as their performance is similar to the best performance by a random ordering We inspected this situation in order to see if there is a systematical error caused by an error during the subjective messiness measurement experiment and found no such error.

In order to ensure the reliability of the subjective messiness information we have collected, we plot the histogram of average index differences between orderings by subjects for a symbol, together with the histogram of average index differences between ordering by a subject and random ordering for a symbol, as well as average index differences between two random orderings for a symbol. Figure 2.10 shows that these two distributions are statistically significantly apart from each other.

In Figure 2.11, the average of the index differences of the all possible subject pairs of each sketch is plotted after the sketches are ordered from neat to messy according to the mean of their rankings by all subjects. A certain trend is visible: sketches with high index difference fall in the middle part of the messiness scale, whereas index difference converges to minimum towards the ends where the sketches are neatest and messiest.

In order to visually demonstrate further how the inter-rater agreement increases as we go to the two ends of the messiness scale, we plotted some additional graphs using

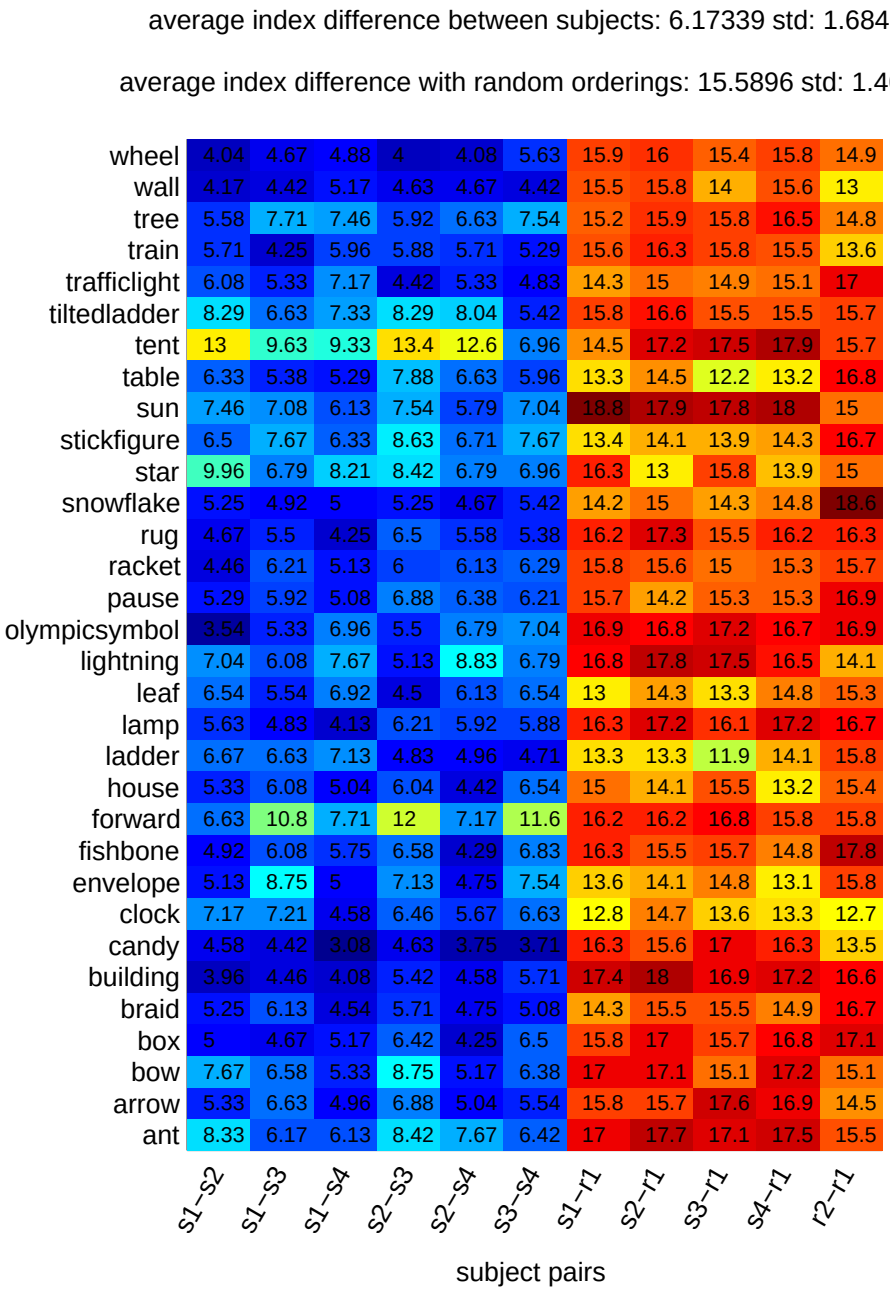


Figure 2.9: Average index differences between the rankings of sketches. s1,s2,s3 and s4 refer to the four subjects who ordered the symbol and r1 and r2 are random orderings that are produced for comparison purposes

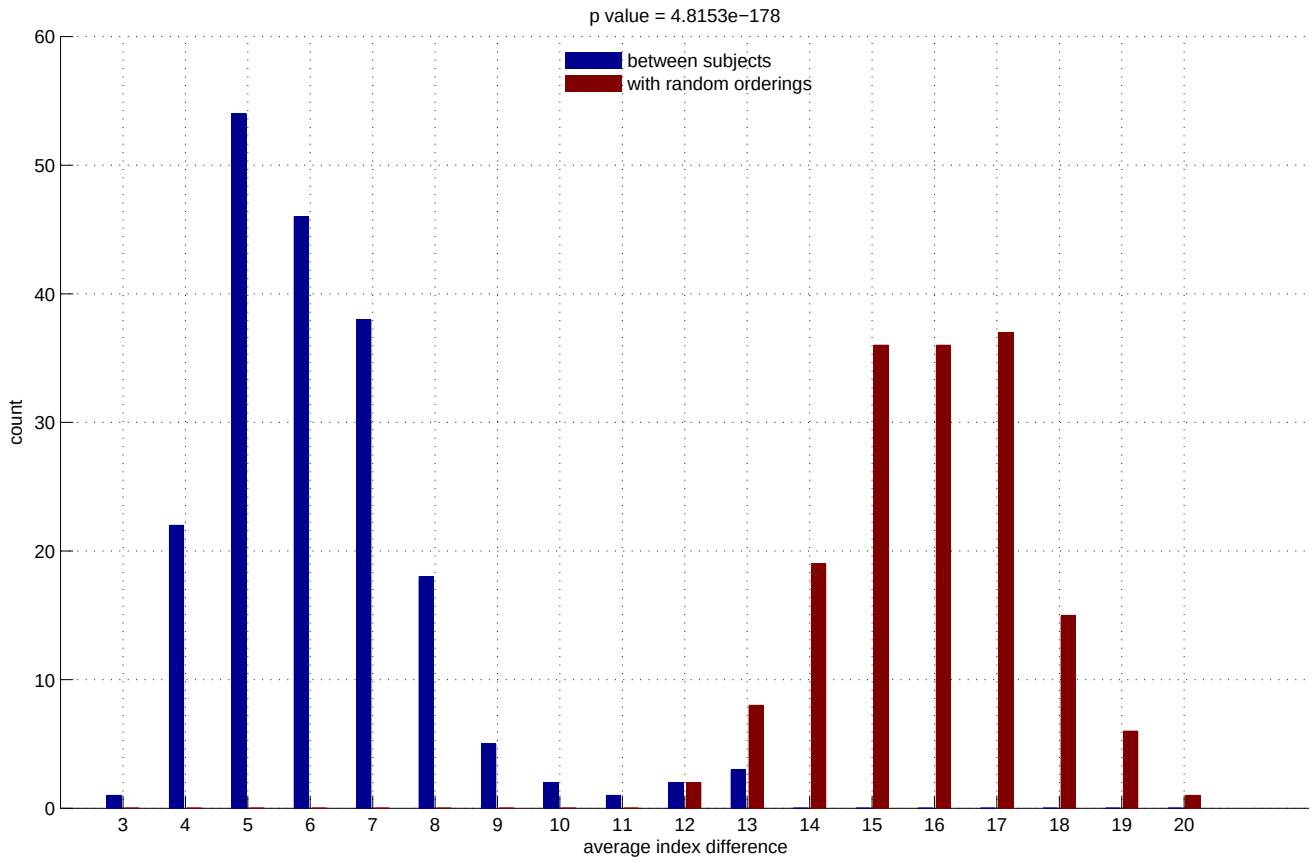


Figure 2.10: Histograms of average index differences between subject-subject pairs and subject-random ordering pairs for a symbol plotted together

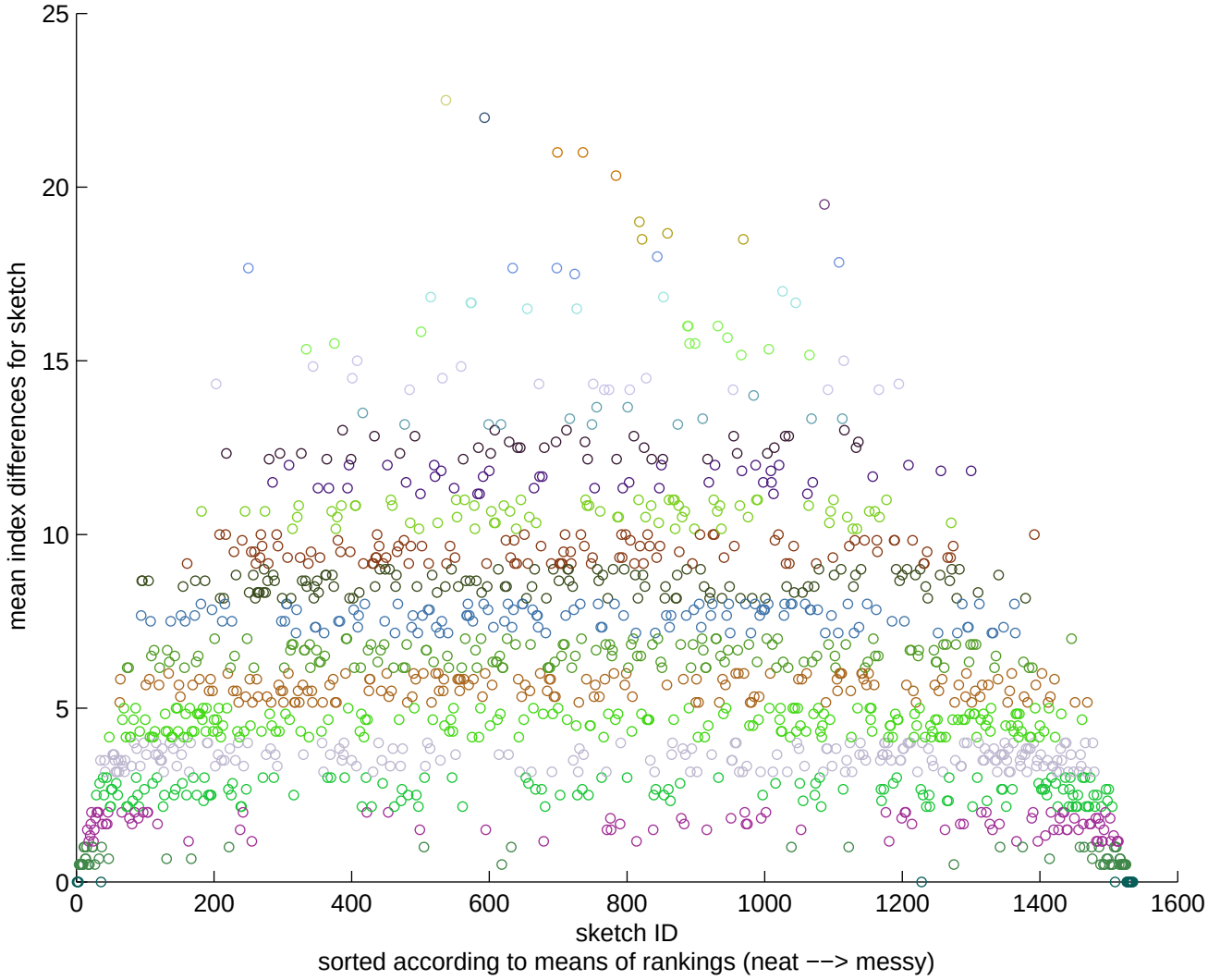


Figure 2.11: Average index differences for each sketch between four subjects who ordered the symbol class of the sketch. Each data point corresponds to a single sketch

the deviation from the mean as metric. Let $\text{rank}(r_n, s)$ be the rank of sketch s by the n^{th} ranker of the symbol that s belongs to, deviation from the mean is calculated as $\text{d}(\text{rank}(r_i, s), mean(\text{rank}(r_1, s), \text{rank}(r_2, s), \text{rank}(r_3, s), \text{rank}(r_4, s)))$ for the i^{th} ranker of s.

Each subplot in Figure 2.12 corresponds to a symbol class and plots how each ranker ranked the sketches of the symbol when the sketches are sorted according to the mean of their rankings. Subplots are also ordered from the symbol with minimum average deviation from mean by its rankers to the symbol with maximum average deviation from mean.

Figure 2.13 shows the deviations from the mean with the same manner in Figure 2.11. The sketches denoted by yellow, green, cyan, blue and pink corresponds to the rating agreement by all the subjects for the rates; very neat, neat, neither messy nor neat, messy, very messy, respectively.

2.2.3 Messiness Scores

After the reliability of the subjective messiness dataset collection is endured by the index difference metric, messiness scores to be used through out the rest of the study is calculated as follows: Let $\text{rank}(s_n, sketch)$ be the rank of sketch $sketch$ by the n^{th} subject who ordered the symbol that s belongs to, messiness score of $sketch$ is calculated as $mean(\text{rank}(s_1, sketch), \text{rank}(s_2, sketch), \text{rank}(s_3, sketch), \text{rank}(s_4, sketch))$.

For example, if a sketch is ranked 4^{th}, 9^{th}, 5^{th} and 6^{th} by the subjects who ordered the symbol class of the sketch, then the sketch's messiness score is computed as: $mean(4, 10, 5, 6) = 6$

Table 2.1 shows the 12 neatest sketches of the dataset in the first two rows, and the messiest in the last two rows. Messiness score of each sketch is shown in the bottom right corner of the sketch.

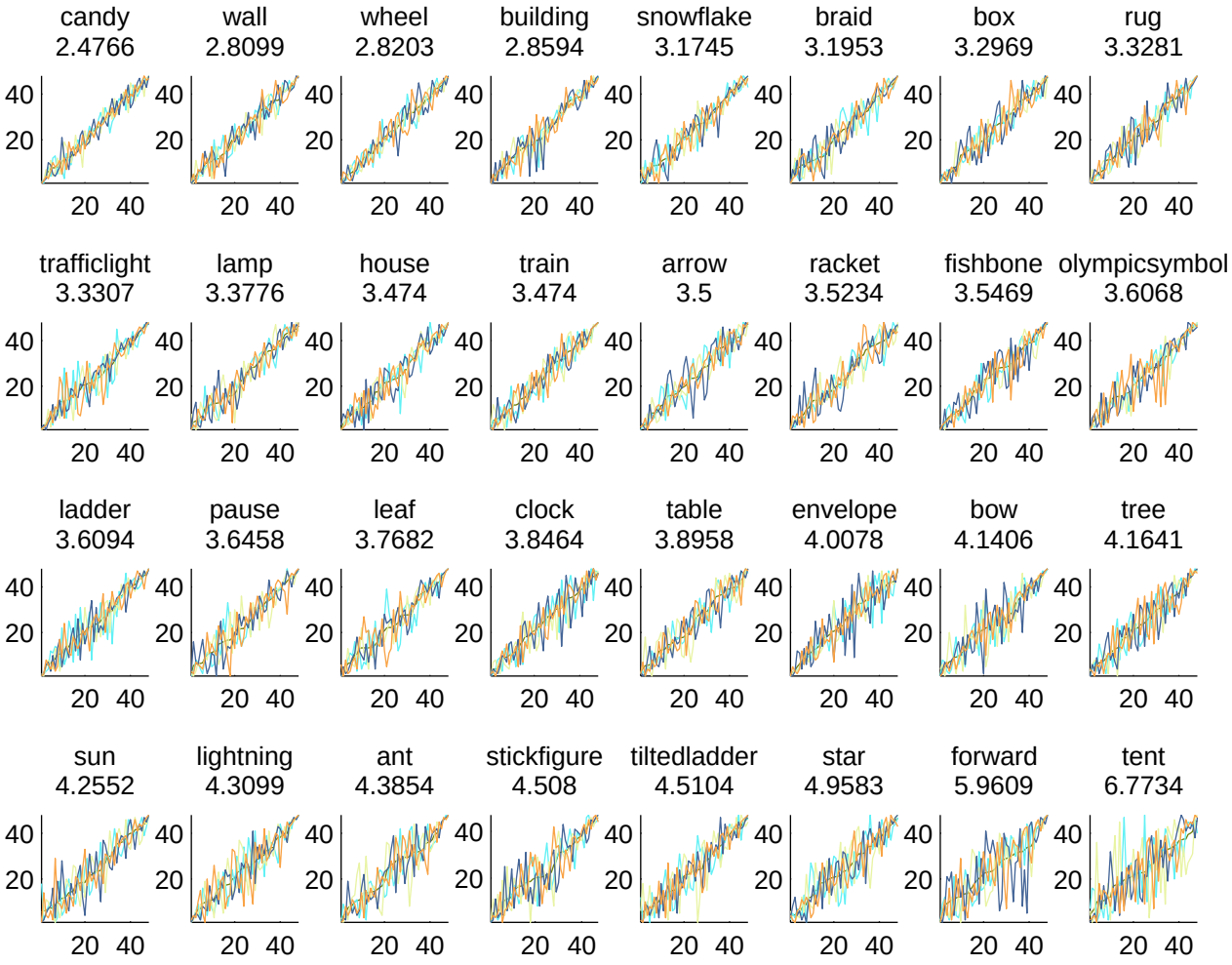


Figure 2.12: Deviations of the ranks for each subject from the mean of ranks. Subplots are sorted so that symbol with least deviation is on the top left and with most deviation is on the bottom right corner.

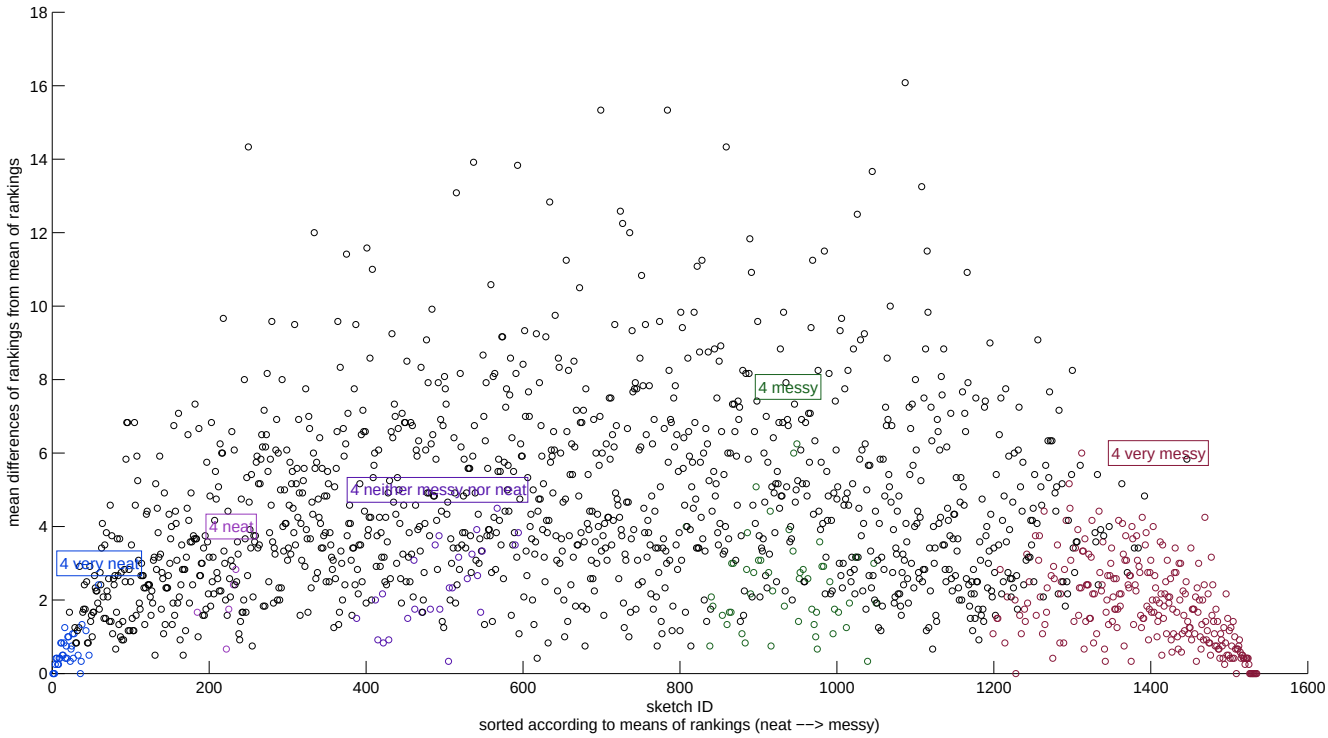


Figure 2.13: Average of deviations of the ranks for each subject from the mean of ranks for each sketch. Each data point corresponds to a single sketch and sketches are sorted according to their mean of ranks

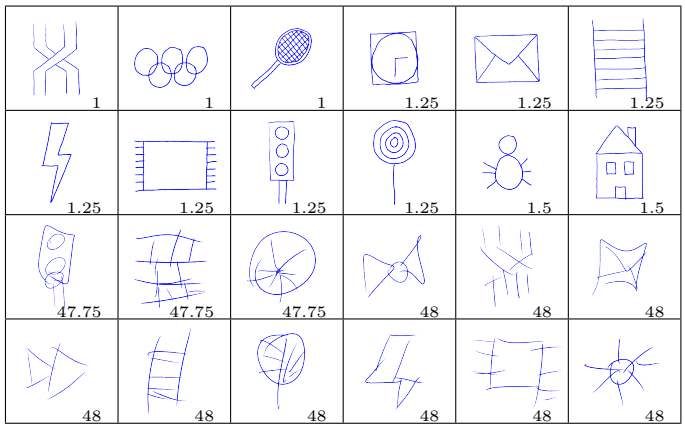


Table 2.1: Neatest and messiest 12 sketches in the dataset with their messiness scores.

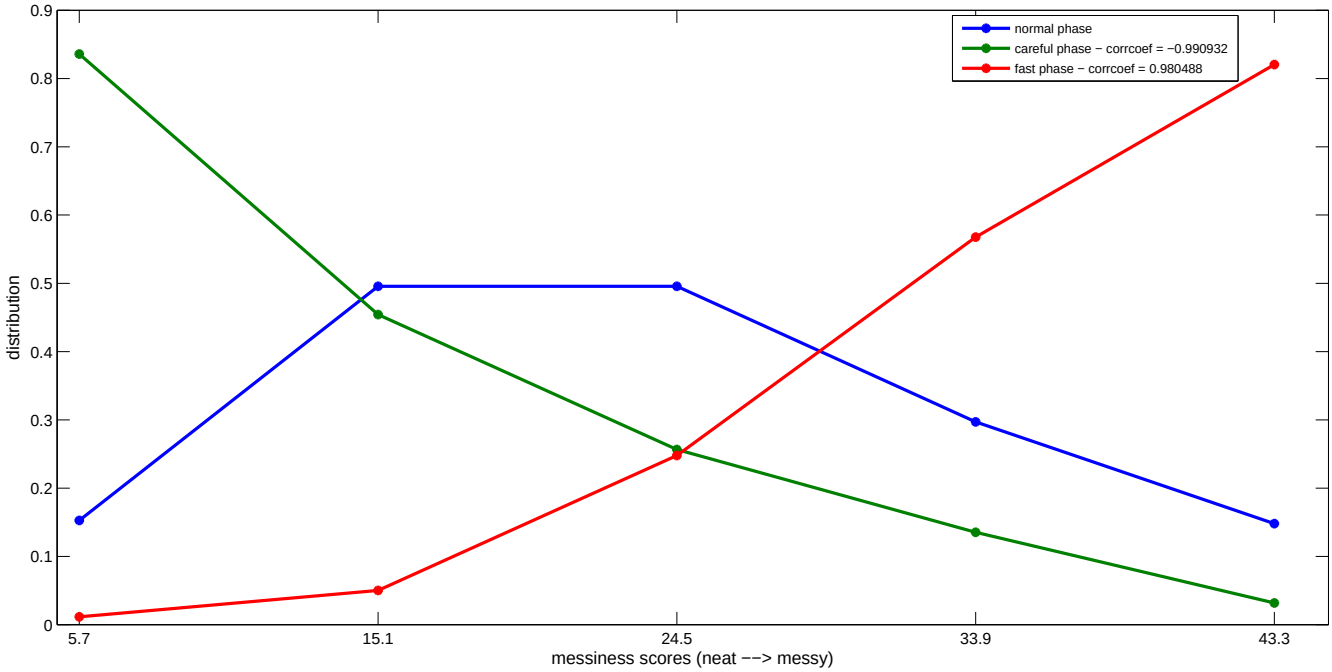


Figure 2.14: Ration of sketches coming from certain phases over five messiness level bins

2.3 Analysis with the Subjective Messiness Dataset

2.3.1 Sketch Data Collection Variables versus Messiness Scores

As our aim was to obtain naturally messy sketches while collecting the sketch dataset via changing restriction on time and diversifying the participants in terms of experience, we expected to see the effect of our efforts on the subjective messiness data.

Figure 2.14 visualizes the strong correlation between sketch coming from careful phase and being neat, as well as sketch coming from fast phase and being messy. On the other hand, highest correlation between age and messiness level comes from the youngest sketcher, as can be seen in Figure 2.15. It is also visible that the designer, who had experience with drawing using tablet and was born in 1987, extensively contributes to the neat end of the messiness scale.

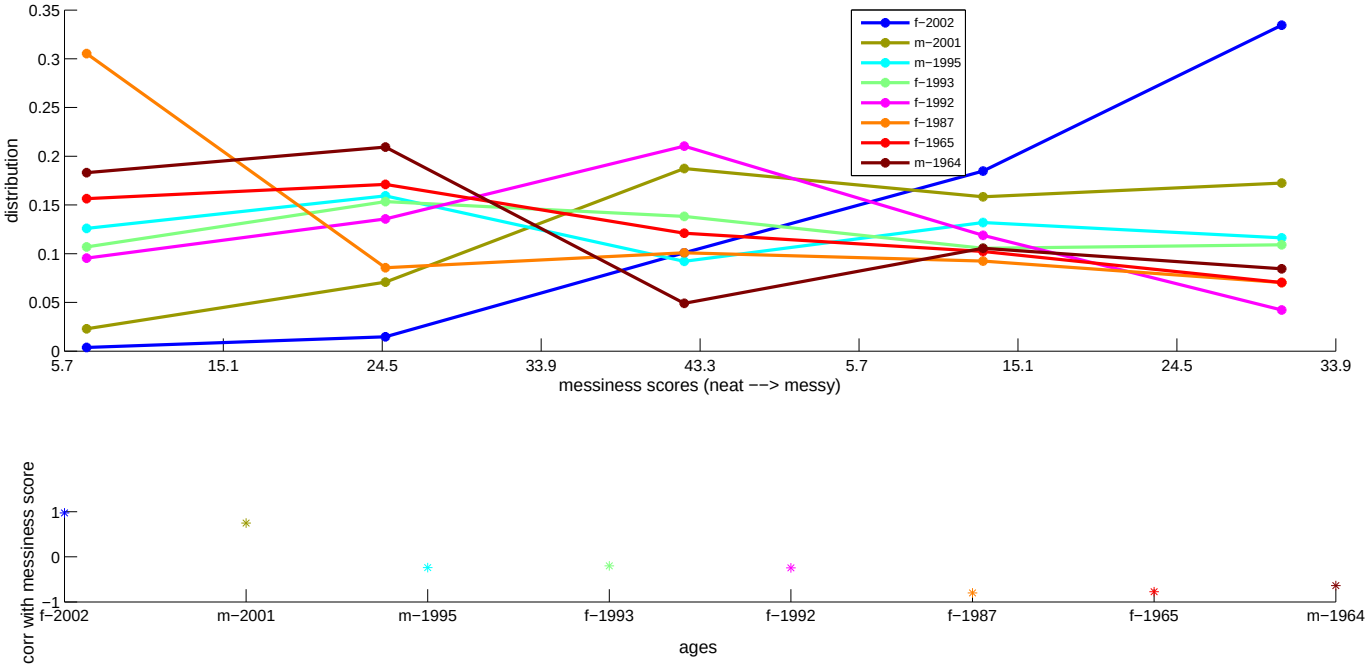


Figure 2.15: Distribution of sketchers (sorted according to age) across messiness level. Designer who participated in our study was born in 1987.

2.3.2 Sketch Recognition Accuracy versus Messiness Scores

Sloppy drawing is listed as one of the factors that reduce reliability of a sketch recognition by [Mahoney and Fromherz, 2002]. Sloppy drawing, here, refers to the situations where a stroke overshoots or undershoots another stroke, where they actually should meet. In order to see if it holds for *messy sketches* that are designated by the participants of our experiment, we inspected the correlation between sketch recognition results and messiness scores of the sketches. Symbol recognition accuracy acquired on our dataset is 79.63%, using SVM classifier ($c = 2^{10}$, $g = 2^{1}$) with Zernike features with the order of 10, which yielded the best accuracy result. Graph at the top of Figure 2.16 shows the mis-recognized sketches in red, with their symbol class and messiness scores. At the bottom of Figure 2.16 is the plot of how missed samples are distributed on messiness scale, when it is divided into 5 bins. As it can be seen in Figure 2.16, number of mis-recognized samples is strongly correlated with the messiness scores of the sketches, with the correlation coefficient of 0.99. Additional analysis is done using IDM features as shown in Figure 2.17, which are proved to be more effective on sketch data for symbol recognition. When IDM is used with the best parameters found after a grid search ($resampleinterval = 4, sigma = 8 and hsize = 3$), with SVM ($c = 2^{3}$, $g = 2^{-10}$) 99.54% accuracy is acquired. The correlation between number of mis-recognized samples per bin, when messiness scale is divided into 5 bins, by IDM features and the messiness scores is still strong, with the correlation coefficient of 0.73.

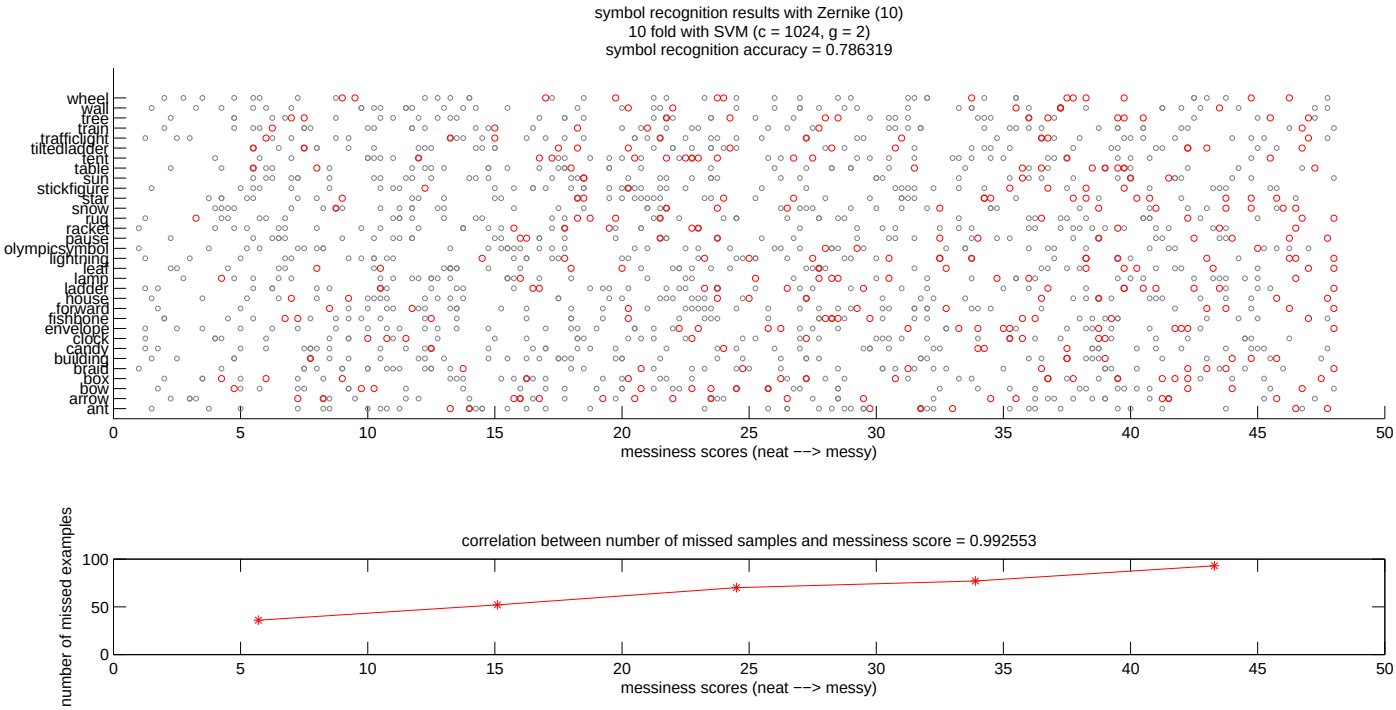


Figure 2.16: Distribution of mis-recognized samples by Zernike features and the correlation of number of mis-recognized samples with messiness scores

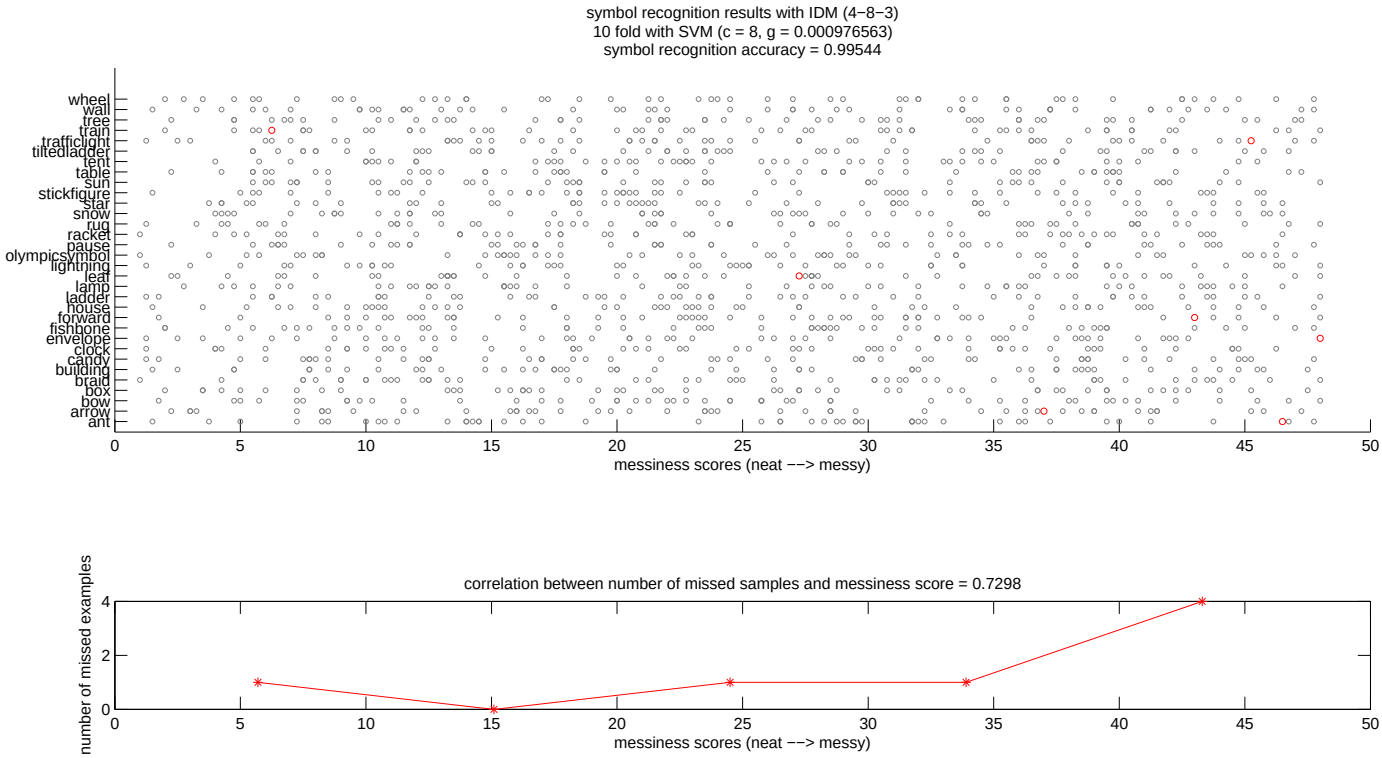


Figure 2.17: Distribution of mis-recognized samples by IDM features and the correlation of number of mis-recognized samples with messiness scores

Chapter 3

FEATURE EXTRACTION AND MESSINESS CLASSIFICATION

Messy sketches may consist of roughly drawn strokes, overshooting or undershooting strokes, as well as geometrical constraints that are intended but not satisfied. Consequently, beautification methods used for computer aided design tools falls into two general categories:

1. Smoothing the strokes so that they are not shaky and/or fractured

2. Ensuring the intended geometrical constraint between the parts of the sketch (e.g. correcting the angle between two lines to make them parallel)

These two categories differ from each other at a very fundamental level: For the former, we can assume that for a sketch to be neat, lines that form the sketch need to be smoothly drawn, thus the smoothness is assumed to be intended. Whereas for the latter, we need the context information, i.e. what is drawn, in order to determine the certain geometrical constraints are intended by the sketcher. Humans can easily use context to intuitively infer if such constraints were intended and use this information when deciding if a sketch is messy or not. Existing beautification tools aims to solve these kind of unknowns via cost functions with thresholds[Paulson and Hammond, 2008] [Murugappan et al., 2009][Igarashi et al., 1997]. For example, two lines are corrected to be parallel if the difference in orientation is less than 3°.

Therefore, we introduce our features in two main categories, corresponding to those mentioned above as seen in Table 3.1.

Regardless of the messiness category, measurements are done for each primitive, i.e. for each primitive in the sketch, primitive fitting error, angle, length and distance

messiness category	feature type
primitive based	primitive fitting error
geometrical measurements	angle
	length
	distance

Table 3.1: Messiness categories and corresponding list of messiness feature types.

are measured. Afterwards, feature extraction is performed on vectors of primitive fitting errors, angles, lengths and distances. The difference between two messiness categories reflects on this feature extraction phase, on how features are extracted from the vectors of primitive measurements.

In this chapter, we introduce our features, how the measurements are done and their corresponding feature extraction methods.

3.1 Primitive Based Features

Most intuitive way of capturing messiness is to calculate how shaky and/or fractured the sketches are. Our metric for measuring these properties of a sketch is what we call *primitive fitting error features*.

3.1.1 Primitive Fitting Error Measurements (fit)

In order to measure the fitting error of a primitive, we first calculate the distances from all the actual sketch points of the primitive to the fitted primitive, as shown in Figure 3.1. Then we take the mean of these distances.

3.1.2 Primitive Fitting Error Feature Extraction

Let v be the vector containing primitive fitting error value for each primitive of the sketch, then the extracted features are as follows:

1. mean(v)

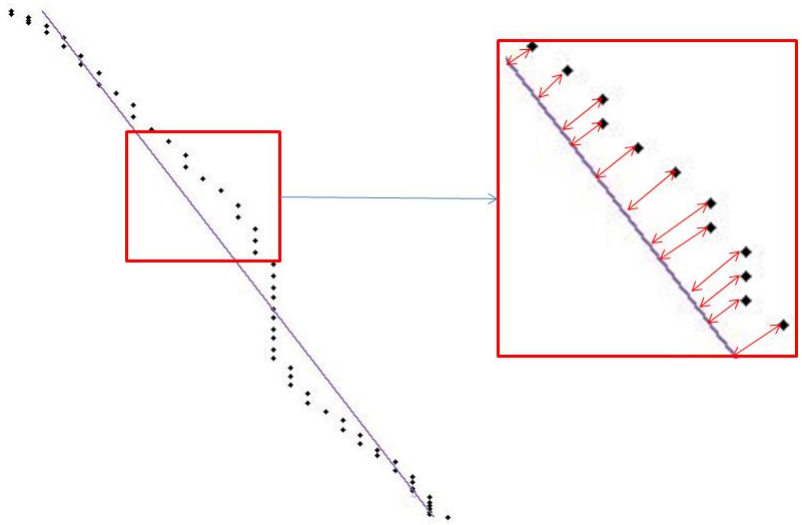


Figure 3.1: Primitive Fitting Error. Black dots correspond to the actual sketch points and blue line is the fitted line.

2. std(v)

3. min(v)

4. max(v)

Feature extraction method for fitting error is different from rest of the feature types, simply because we assume that the fitting error need to minimized in order to obtain neat sketches.

3.2 Geometrical Measurements Features

Geometrical constraints are expressed using measurements such as angle, length and distance. For examples, for two lines to be equal, *angles* and *lengths* of the lines must be the same. Most common measurements yielded to be these two, according to our literature survey on geometrical constraints. In order to utilize the calculation of deviation from certain geometrical constraints for predicting messiness, we first needed the information on if the constraint in question is actually intended. This

requires either the context information of the sketch, i.e. the information on what is drawn so that we *know* which constraints are intended, or the use of thresholds, as beautification systems generally depend on. We aimed our system to be free of thresholds in order to make it generalizable. During our search for such a solution, we produced an informative figure showing how the distribution of angles changes for a symbol with the level of messiness as can be seen in Figure 3.2.

Figure 3.2 each line represents one of the sketches of the building symbol. There are 48 lines for all sketches of building class. Neatest building sketch, shown on the bottom left corner, being the left most,whereas the messiest, shown on the bottom right, being the right most line. Each circle on a line represents a single measurement belonging to a primitive of the sketch. The figure clearly shows that the building symbol has two clusters of angle values, and this clusters are more dense towards the neat end. A neat building sketch consists of horizontal and vertical lines, measuring 0° and 90°, respectively.

But how do we capture these clusters, without the context information (e.g. if the symbol is building, we have two clusters). Section 3.2.4 explains our method, after the feature types are explained.

3.2.1 Angle Measurements (angle)

Angle measurements are for the orientation of the primitives, so that the parallelism and perpendicularity can be captured. It is calculated as shown in 3.3:

- For lines: (1)acute angle of a line passing through start and end points of the primitive to horizontal.

- For ellipses: (1)acute angle of line passing through the points with the minimum and maximum y values.

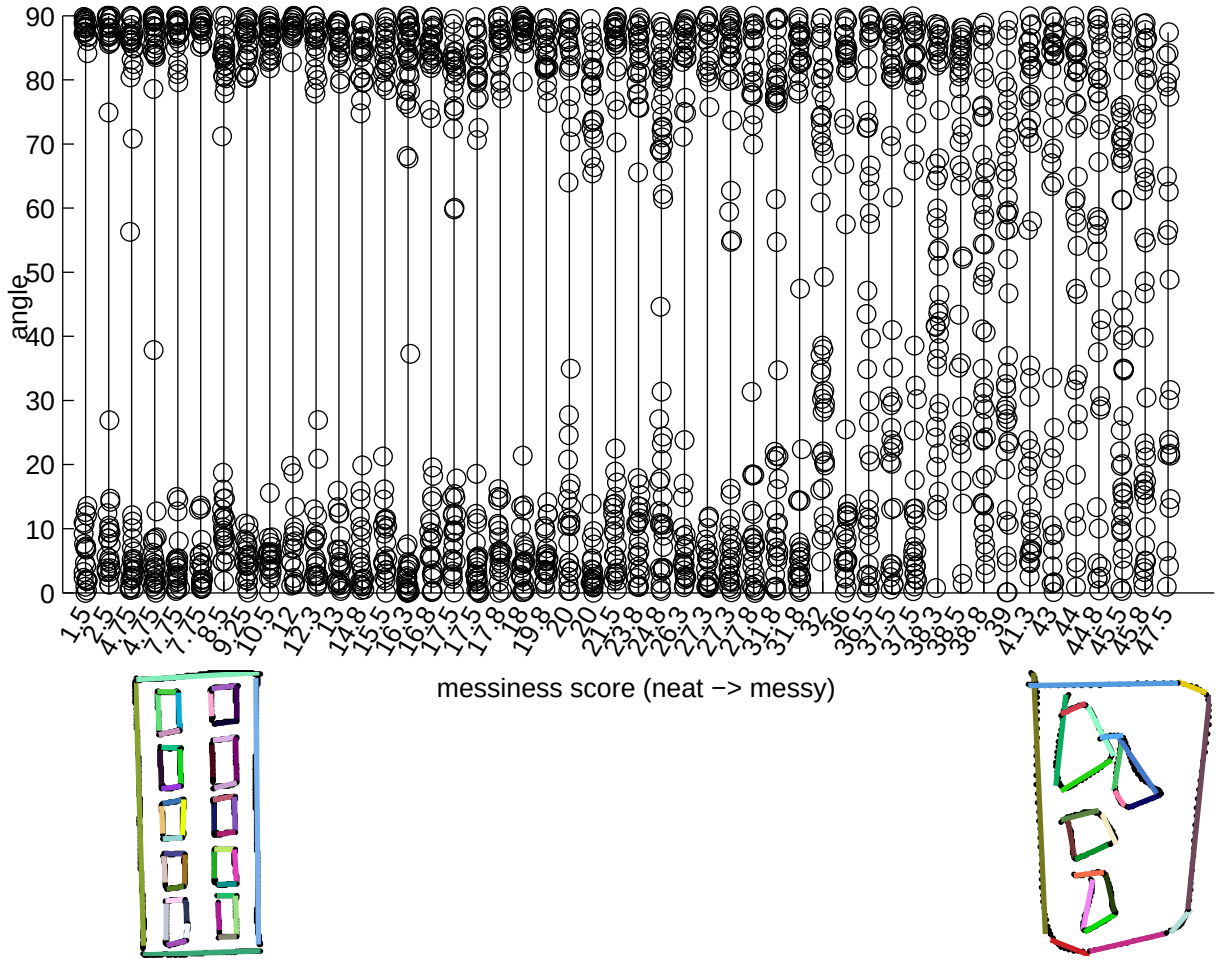


Figure 3.2: Angle measurements the building symbol.

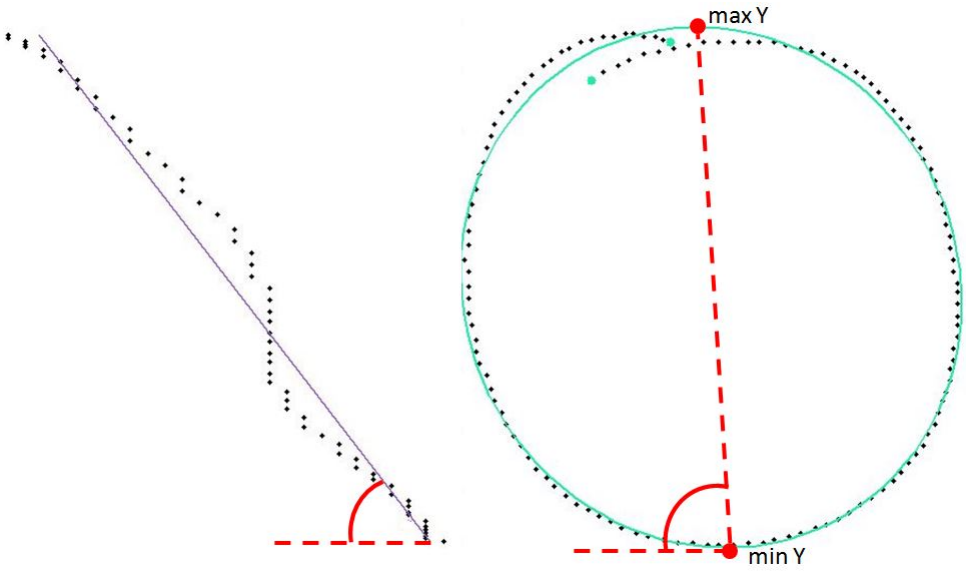


Figure 3.3: Angle measurements for line and ellipse

3.2.2 Length Measurements (length)

Length measurements are to find the patterns of similar length primitives in the sketch, e.g a square is as neat as how similar in length are its lines.

- For lines: (1)Euclidean distance between the start and end points.
- For ellipses: (1)Euclidean distance between points with minimum and maximum x values and (2)Euclidean distance between points with minimum and maximum y values.

Figure 3.4 shows how the lengths are measured by two headed arrows.

3.2.3 Distance Measurements (distance)

Distance measurement is similar to length in nature, i.e. both calculated as Euclidean distance, but the aim is to detect [Mahoney and Fromherz, 2002]'s *sloppy drawing*: Does a primitive overshoot or undershoot another primitive, where they actually should meet. The information of *whether they should meet* is again the matter of context issue. We need the context information on sketch, in order to determine if

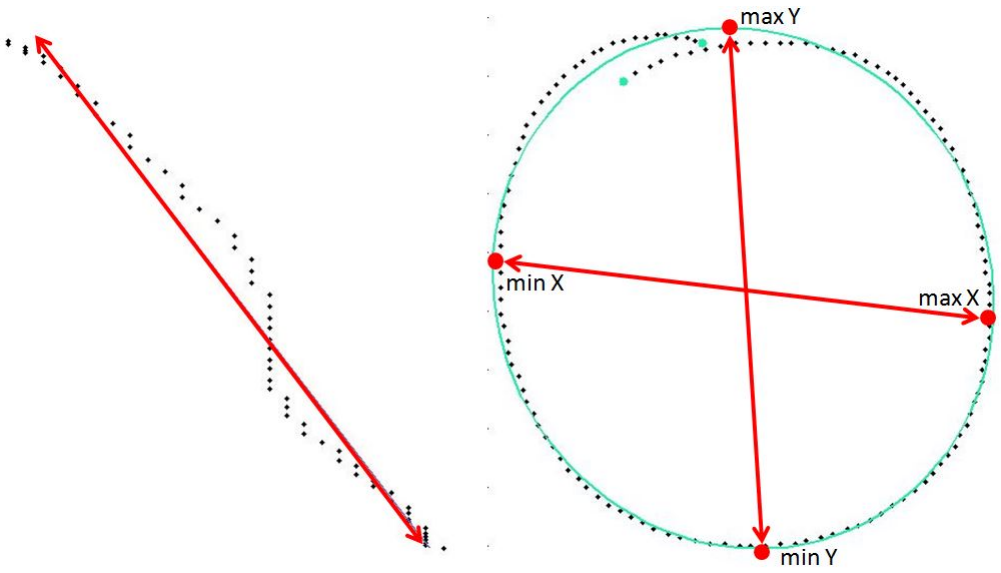


Figure 3.4: Length measurements for line and ellipse. Calculated lengths are denoted by two headed arrows

the primitives of sketch are to meet. Our intention is to bypass this context issue by considering only one closest point to each primitive end-point (first and last points of a primitive). Since we restricted the primitive types to ellipse and line, we assume that end-points of each ellipse are to touch each other. On the other hand, the line end-points are to meet (if they are to meet any primitive) a point from a primitive, other then itself. Therefore distances are calculated as:

- For lines: (1)Euclidean distance between the start point and the closes point belonging to another primitive of the sketch and (2)Euclidean distance between the end point if line and the closes point belonging to another primitive of the sketch.

- For ellipses: (1)Euclidean distance between the start and end points.

Please see Figure 3.5 for the visualization of how distances are measured.

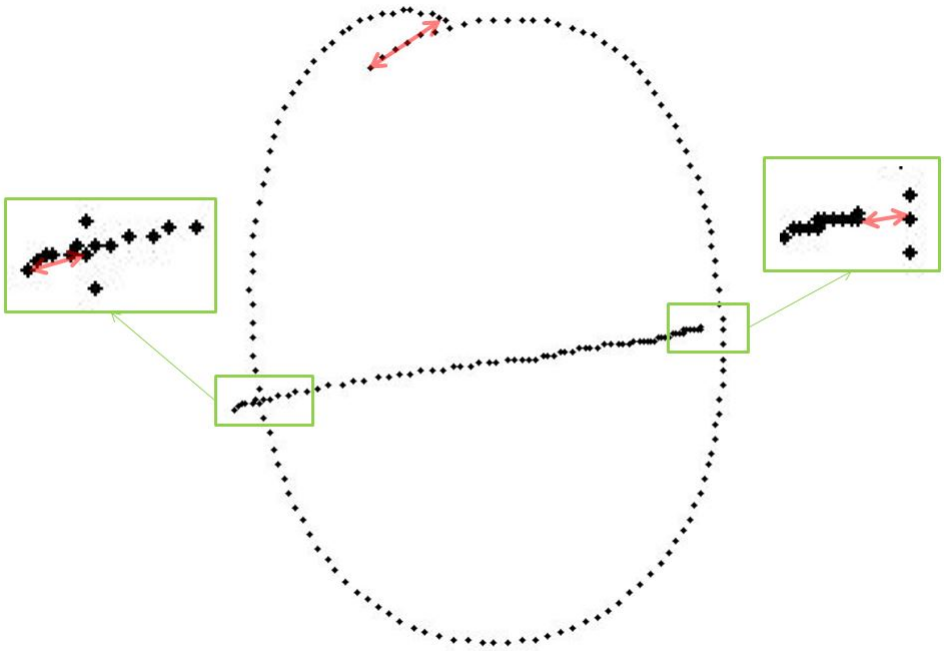


Figure 3.5: Distance measurements for line and ellipse. It is the distance between the end-points of an ellipse, and the distance between closest point from another primitive to the each end-point of a line

3.2.4 Geometrical Measurements Feature Extraction

For a sketch consisted of n primitives, let v be the vector of size $1xn$ containing one type of the geometrical measurement (angle, length, distance) value for each primitive of the sketch. Features are extracted as follows:

- v is sorted in ascending order of the values
- $v_{sorted} =< m_1, m_2, m_3..., m_n >$ is obtained, so that $min(v) = m_1$ and $max(v) = m_n$
- difference between successive points in v is calculated as: $sd =< m_2 - m_1, m_3 - m_2, ..., m_n - m_{n-1} >$
- sd is clustered into 2 clusters, c_1 and c_2, using kmeans clustering function from MathWorks statistical toolbox, using squared Euclidean as distance measure-

ments and setting 'emptyaction' to 'singleton' which makes sure that a second cluster is formed even if it contains a single data point.

Then the final list of extracted features is:

1. $center_{c_1}$
2. $\text{mean}(|points_{c_1} - center_{c_1}|)$
3. $\frac{\text{number of points in } c_1}{\text{number of points in sketch}}$
4. $|center_{c_1} - center_{c2}|$
5. $\text{mean}(|points_{c2} - center_{c2}|)$
6. $\frac{\text{number of points in } c_2}{\text{number of points in sketch}}$

Our approach would be to cluster the measurements vector into suitable number of clusters if we had the context information of the sketches before hand, e.g. there are two visible clusters in the angles distribution of building symbol as seen in Figure 3.2. Since we do not have the symbol class information on sketches, our aim while we cluster the successive differences of the sorted measurements vector into two clusters is to detect the intra-group variance and inter-group distance of the visible clusters on the initial measurements distribution.

3.3 Features merged

Table 3.2 sums all feature types with its measurements and feature extraction methods. The vectors or points obtained in the first column is combined with the calculation in the third column for each primitive. When calculations of each primitive is completed for a feature type, then the vector of values coming from these calculations are transformed into features using the corresponding method in the last column. Feature extraction on sample sketches can be seen in Figure 3.6, for a need example

<table>
<tr><td></td><td colspan="3">measurement - primitive level</td><td>feature extraction - sketch level</td></tr>
<tr><td></td><td>line</td><td>ellipse</td><td>calculation</td><td></td></tr>
<tr><td>fit</td><td colspan="2">v_1 :sketch points
v_2 :corresponding closest points from fitted primitive</td><td>mean(d(v_1, v_2))</td><td>(1) mean($measurements$)
(2) std($measurements$)
(3) min($measurements$)
(4) max($measurements$)</td></tr>
<tr><td>angle</td><td>$p_1 : p_{start}$
$p_2 : p_{end}$</td><td>$p_1 : p_{minY}$
$p_2 : p_{maxY}$</td><td>$\measuredangle\overrightarrow{p_1p_2}$</td><td rowspan="4">$sd$ = differences(sort($measurements$))
cluster($sd, 2$) $\rightarrow c_1, c_2$:
(1)$center_{c_1}$
(2) mean($|points_{c_1} - center_{c_1}|$)
(3)$\frac{\text{number of points in } c_1}{\text{number of points in sketch}}$
(4)$|center_{c_1} - center_{c2}|$
(5) mean($|points_{c2} - center_{c2}|$)
(6)$\frac{\text{number of points in } c_2}{\text{number of points in sketch}}$</td></tr>
<tr><td>length</td><td>$p_1 : p_{start}$
$p_2 : p_{end}$</td><td>$p_1 : p_{minX}$
$p_2 : p_{maxX}$
$p_1 : p_{minY}$
$p_2 : p_{maxY}$</td><td>d(p_1, p_2)</td></tr>
<tr><td>distance</td><td>$p_1 : p_{start}$
$p_2 : p_{\text{closest}}$
$p_1 : p_{end}$
$p_2 : p_{\text{closest}}$</td><td>$p_1 : p_{start}$
$p_2 : p_{end}$</td><td></td></tr>
</table>

Table 3.2: Feature types together with their extraction methods

from racket symbol, and 3.7, for a messy example from train symbol. In the figures, fitting error features are shown as values but geometrical measurements feature extraction visualization is left at the clustering level.

3.3.1 Measurements across messiness

Figure 3.8 shows how the distribution of the measurements change with messiness. We report this for the sketches of same symbol, as distributions are context dependent. For each sub figure, each line represents one of the sketches of the symbol in question. There are 48 lines for all sketches of the symbol; neatest sketch of the symbol, shown on the bottom left corner, being the left most and the messiest, shown on the bottom right, being the right most line. Each circle on a line represents a single measurement

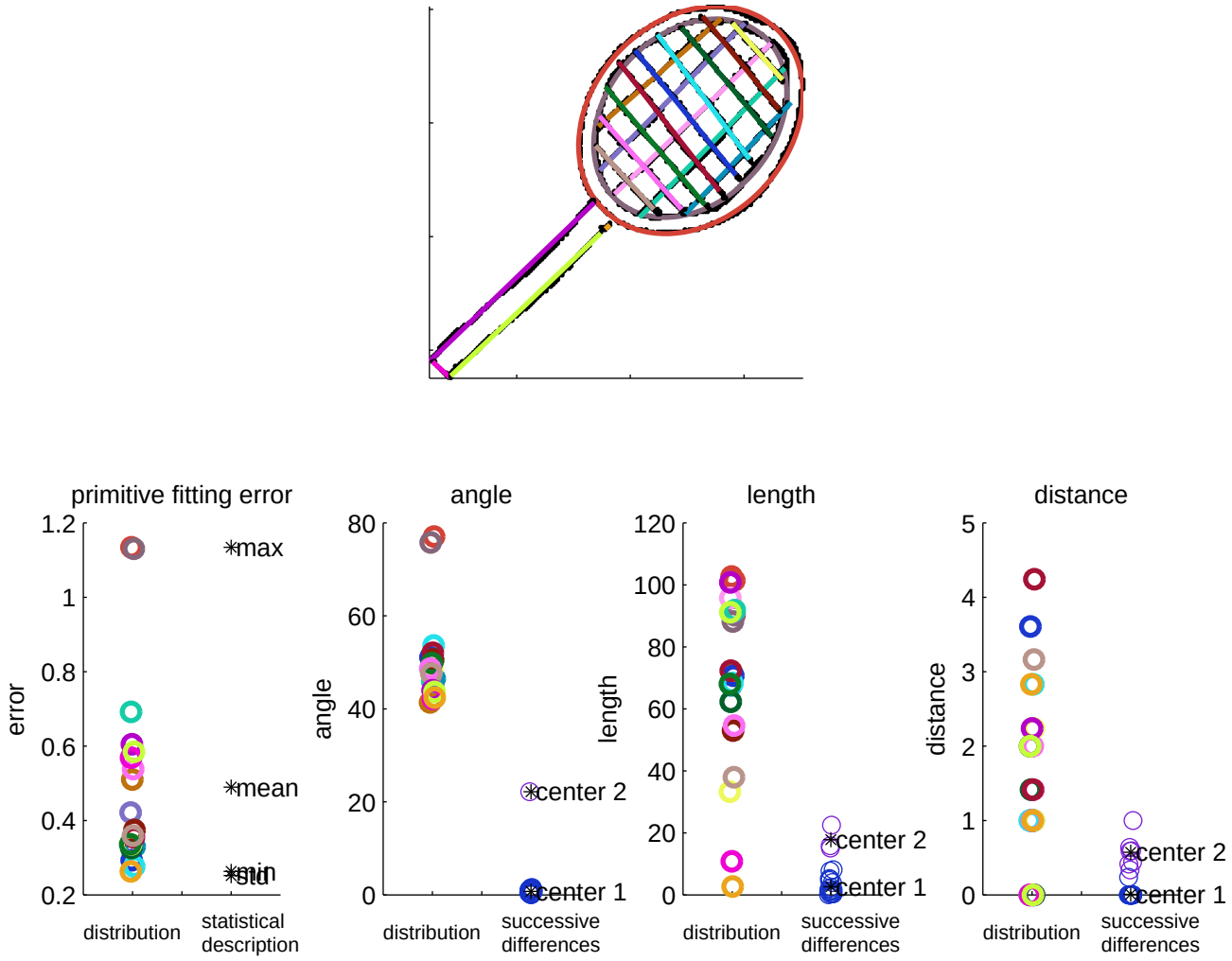


Figure 3.6: Feature extraction for a neat sketch from racket symbol

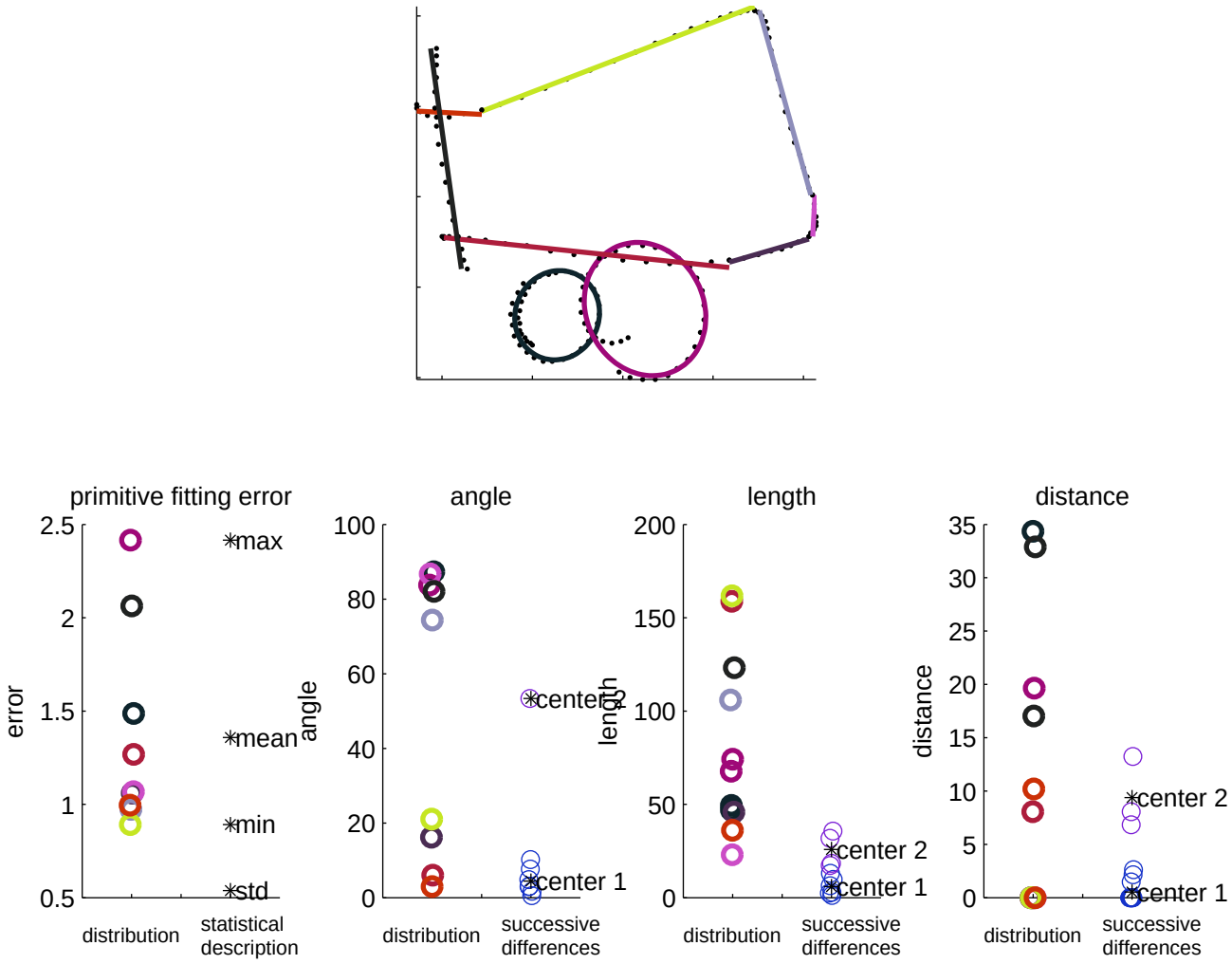
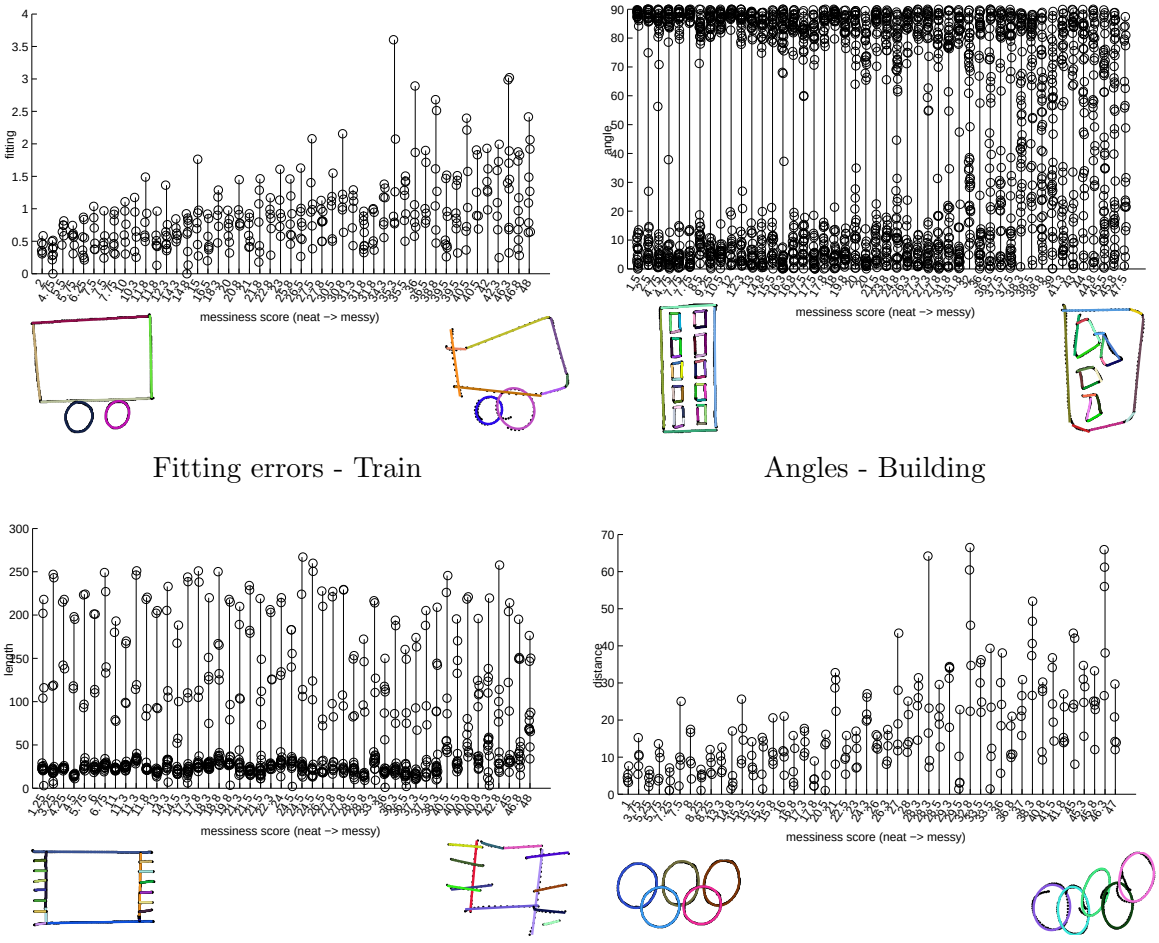


Figure 3.7: Feature extraction for a messy sketch from train symbol



Fitting errors - Train

Angles - Building

Lengths - Rug

Distances - Olympicsymbol

Figure 3.8: Measurements for features of all sketches of a single symbol.

belonging to a primitive of the corresponding sketch. The increase in the fitting errors is visible as the messiness score of the sketches increase in Figure 3.8i. Angles start as two distinct clusters from the neat end, and the distinction lessens towards the messy end. Lengths for rug symbol concentrate between 0 and 50, but this concentration spreads closer to the messy end. Similarly, two clusters are visible in the neat part of olympicsymbol, but the clusters disappear towards the end.

3.4 Classification Method

For the classification purposes, we used multi class Support Vector Machine (SVM) classifiers with the parameters: $c = 2^{10}$, $g = 2^{-3}$, decided after parameter grid search.

Chapter 4

MESSINESS CLASSIFICATION EXPERIMENTS

This chapter inspects the relation between messiness prediction accuracy and

1. various combinations of feature types (fit,angle,length,distance)

2. various dataset size

3. various messiness scale of the dataset

4. various levels of subject agreement on messiness

and the combinations of these on the messiness prediction accuracy.

Each experiment is a 2-class (*neat* and *messy*) classification problem, performed for all the combinations of feature types. We report the results for the feature type combinations that demonstrate the additional boost achieved by adding geometrical measurements features on top of the primitive based features. This is simply because fitting error features are quite strong on their own, improving them has become our objective as the study progressed.

4.1 Subject agreement on messiness

Level of subject agreement highly correlates with the messiness score, as we mentioned in Section 2.3. Therefore, we perform 5-fold SVM classification and record the prediction result, $true$ or $false$, for each sketch. 5-fold classification gives us a training set of 1228 sketches picked uniformly in terms of their messiness scale as shown in Figure 4.1.

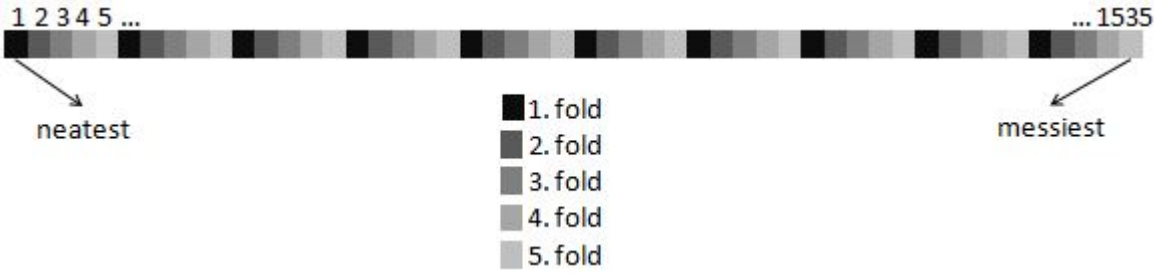


Figure 4.1: Uniform 5-folding

After the prediction results for each sketch is obtained, in Figure 4.2, we plot how correctly predicted samples distribute over the messiness scale.

Figure 4.2 shows how different types of features perform on different parts of the messiness scale and also level of agreement between subjects. Each sketch in dataset has a prediction result obtained.

4.2 Dataset size

Increasing the training set size is a method proven to increase the accuracy result. In order to see the effect of the increased dataset size, we first divided the whole dataset into 10 folds, in a similar manner in Figure 4.1. Then the experiment is conducted as in Algorithm 1.

Figure 4.3 shows the increase in messiness prediction accuracy as the training dataset size increases, for different combination of features types.

For further analysis on dataset size, we add one more variable to the equation, which is the messiness level distinction between the messy and neat classes, called *messiness distance*. Our intuition was that further the neat and messy classes are on the messiness scale, i.e. greater the messiness distance, better the accuracy is. For this purpose, for each dataset size, n, we formed the dataset for the experiment by taking the $n/2$ neatest and $n/2$ messiest sketches. As it can be observed from Figure 4.4, as the dataset size increases, messiness distance between neat and messy classes decreases. Figure 4.5 shows the decrease in messiness prediction accuracy even though

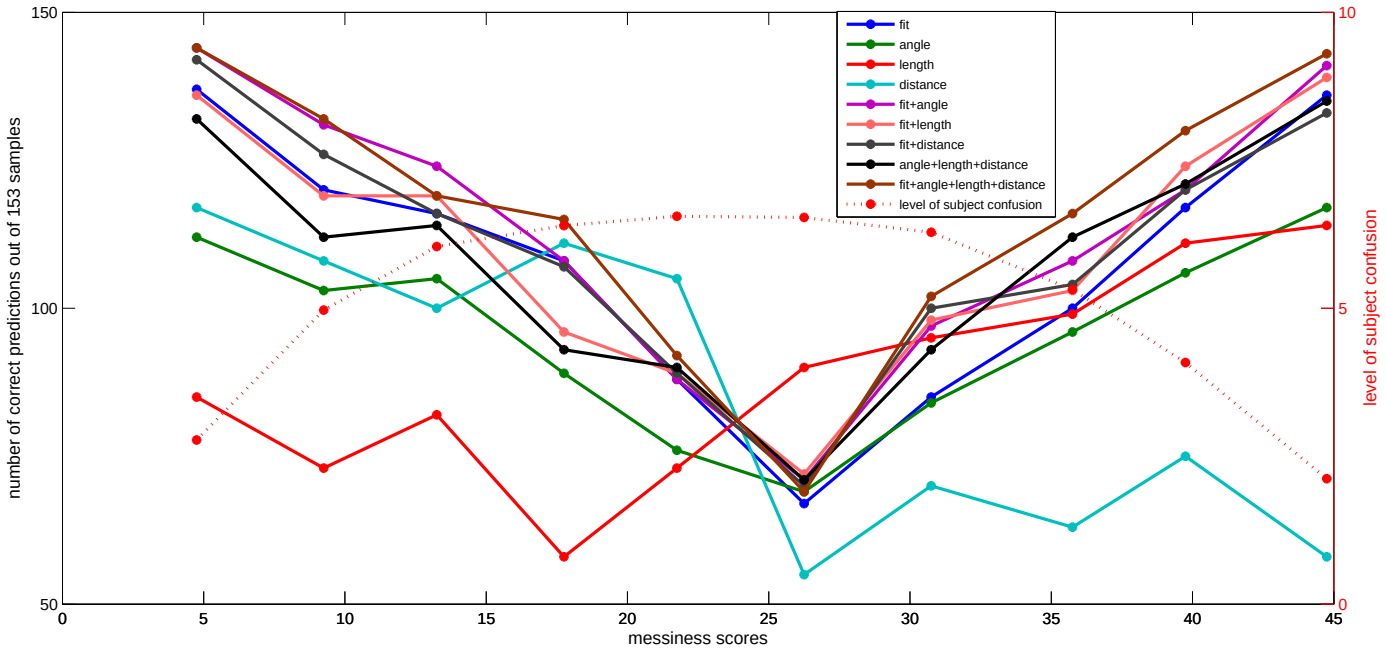


Figure 4.2: Number of correct predictions vs. subject agreement and messiness level

Algorithm 1 Various dataset sizes experiment

$N \leftarrow 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000$
for $f = fold_1$ to $fold_{10}$ **do**
  $test\ set \leftarrow f$
  for $shuffle = 1$ to 10 **do**
    $training\ pool \leftarrow$ randomize the remaining of the dataset
    for all $n \in N$ **do**
      $train\ set \leftarrow$ first n sketches from trainingpool
      $accuracy \leftarrow$ SVM classification with $c = 1024$ and $g = 0.125$ on $train\ set$ and $test\ set$
      return $accuracy$
    end for
  end for
end for

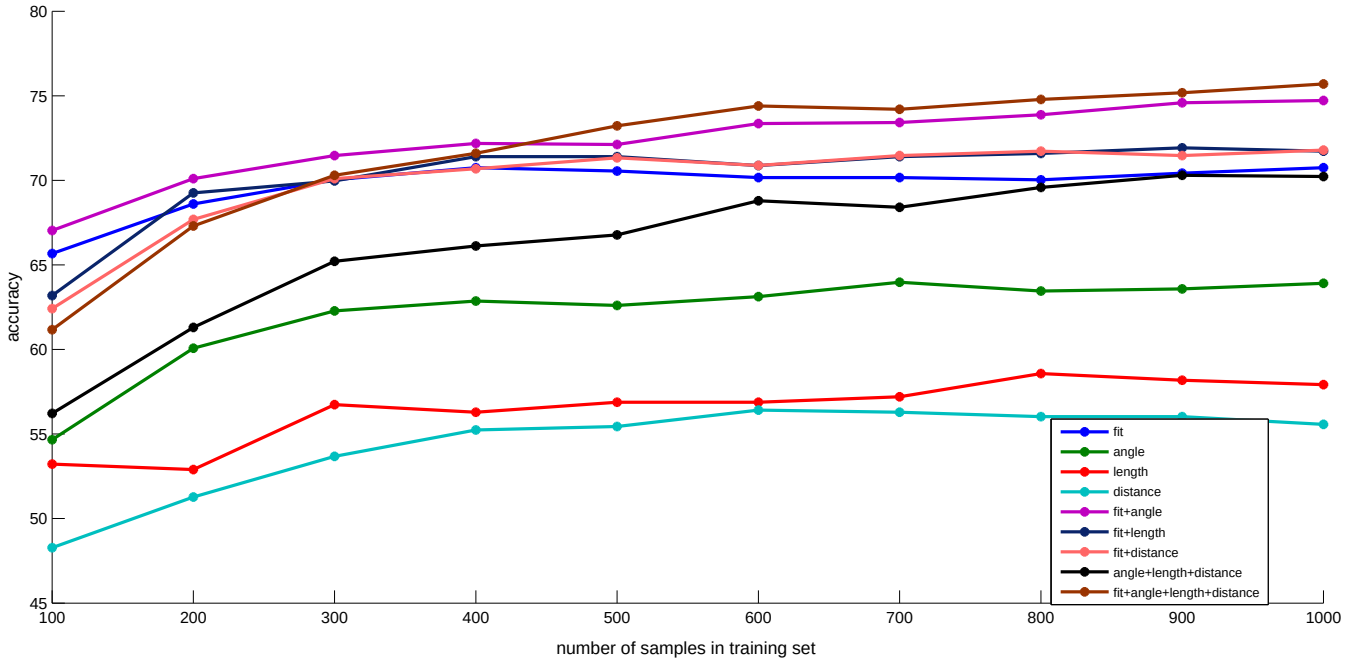


Figure 4.3: Classification accuracies for various sizes of training set

the training dataset size increases due to the decrease in the messiness distance .

4.3 Messiness distance

As decrease in messiness distance, canceled out the effect of the increase in dataset size, we decided to conduct an experiment which we only vary the messiness distance between the two classes, neat and messy.

Whole sketch dataset is divided into 10 folds after it is sorted according to the messiness scores, as it can be seen in Figure 4.6. We compare the classification accuracies for every possible pair of folds as the messiness classes and report the results. Messiness distance between classes increase, as the difference between fold numbers increase. Messiness prediction accuracies for three different combination of features are shown in Figure 4.7. The feature combinations reported are (1)fitting error features, (2)angle + length + distance features, (3)all features. A clear pattern is

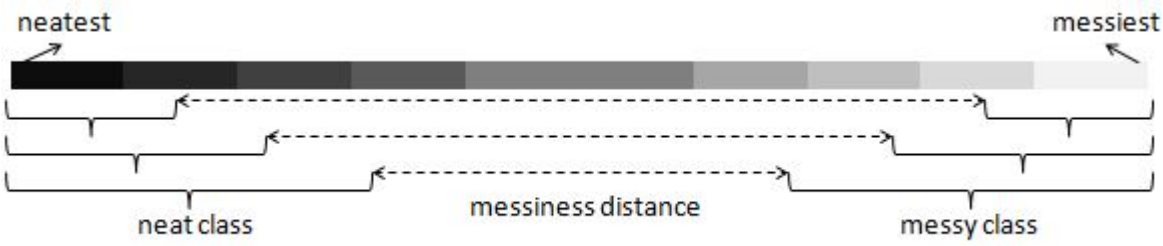


Figure 4.4: Messiness classes formed by taking samples from the neatest and messiest end of the scale

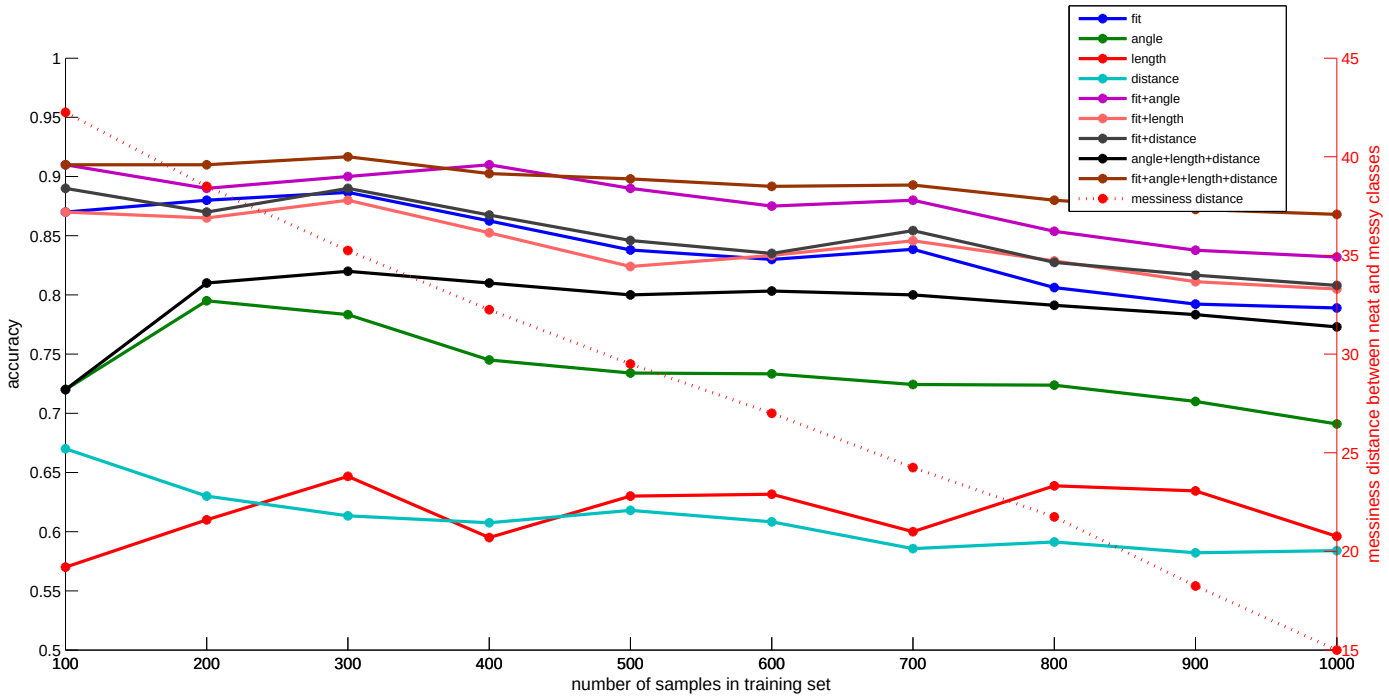


Figure 4.5: Classification accuracies for various dataset sizes and messiness distances. Right y-axis corresponds to the dotted line.

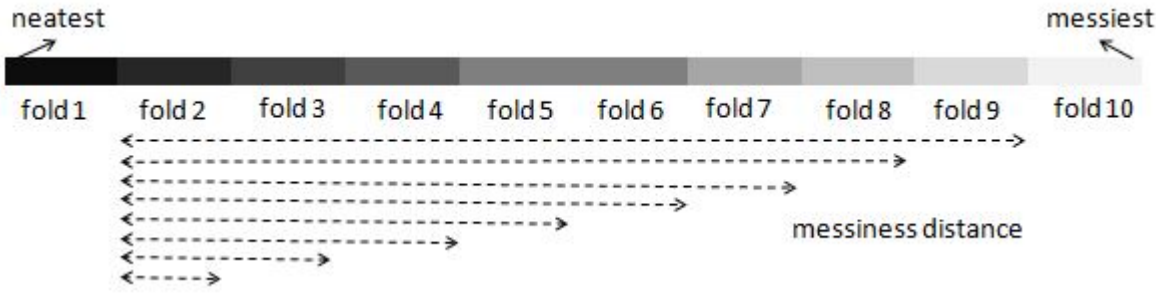


Figure 4.6: Messiness classes formed by 10 folds with messiness distance

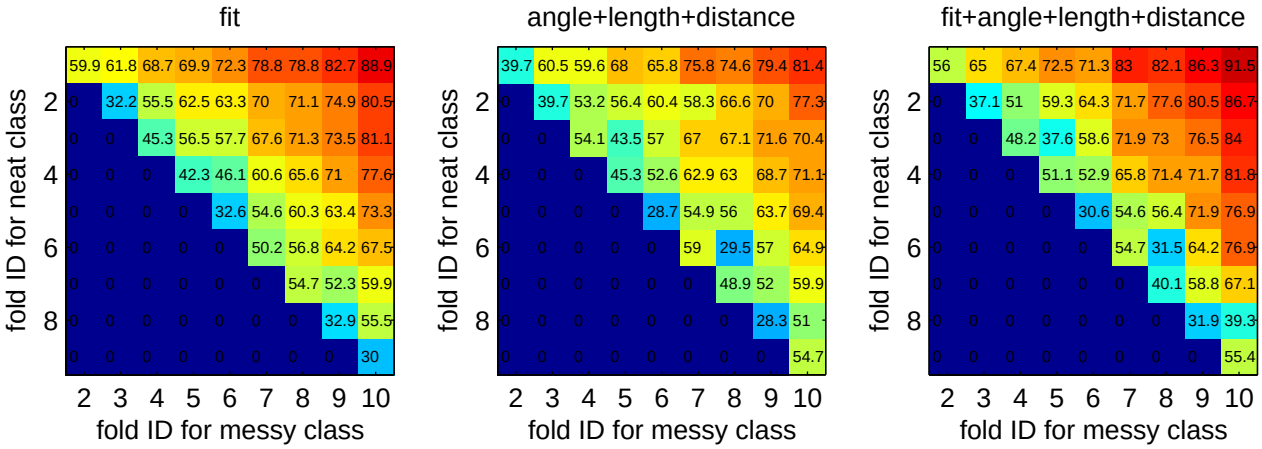


Figure 4.7: Messiness prediction accuracies according to messiness distance between classes

visible for all three of them: As the messiness distance increase, messiness prediction accuracies increase also.

4.4 Generalizability

Generalizability of our system can be measured by how well messiness prediction works for unseen symbols. For this purpose, we designated the messiest 100 and neatest 100 of the whole dataset so that the subject disagreement is minimized, while keeping the dataset size moderate. Figure 4.8 shows the results for each symbol. Best performing combination of features for this purpose, with the average accuracy of

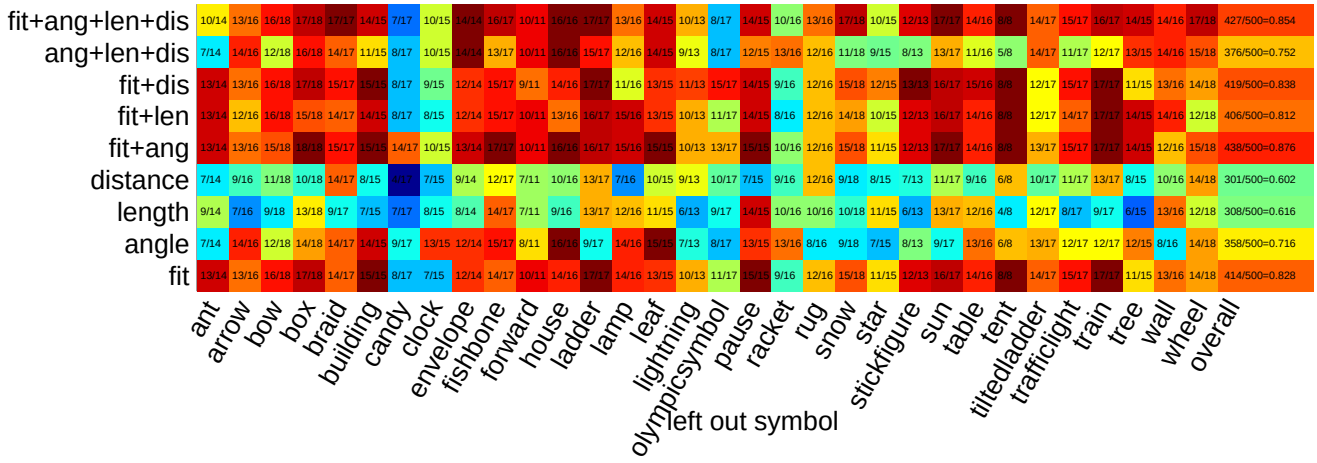


Figure 4.8: Accuracies of predicting the messiness class of unseen symbol

87.6%, is primitive fitting error features combined with angle features.

Chapter 5

CONCLUSIONS AND FUTURE WORK

In this thesis, we introduced a novel problem of predicting the messiness class of hand-drawn sketches as perceived by humans. We demonstrated the difficulties of this problem on a large sketch dataset we have collected with subjective messiness information on each sketch. We also reported where the sketches of sketchers with various backgrounds and ages fall on the messiness scale.

Most intuitive messiness features performed quite well as expected: A metric called primitive fitting error, measuring how much the sketch primitives deviates from perfect ellipses and lines. We were able to improve the primitive fitting error features, which proved to be a strong group of features, with features based on geometrical measurements, such as orientation, length and distance. We avoided utilizing the context information at any stage.

Our messiness prediction accuracies varied depending on various aspects of the dataset that is used during classification. We report how accuracies change by the training dataset size, level of subject agreements on messiness and messiness distance between the two messiness classes, neat and messy. Our method performed substantially better when there was high consensus on the messiness class of sketches. Overall, our approach achieved over 90% accuracy on the subpart of the dataset with a moderate size and high consensus.

5.1 Future Work

We plan to continue our experiments for further analysis on the size of the training set, when the system is to predict messiness classes for samples of an unseen symbol. Another analysis is to be done on how our system performs when the dataset consists

of a single symbol class. We also plan to extend this study to see how accurately would our system do pairwise messiness comparisons of sketches.

We would also like to test the system's generalizability, by predicting messiness classes of a completely different dataset with subjective messiness information.

In the future, we plan to improve our features so that they achieve sufficient performance when the messiness prediction problem is treated as a classification problem with more classes, or even as a regression problem. We hope to be able to modify our system so that it learns the people's preference when assessing its aesthetics, i.e. which properties of a sketch is more important for a sketch to be perceived as a neat one.

BIBLIOGRAPHY

[Adali et al., 2007] Adali, S., Magdon-Ismail, M., and Marshall, B. (2007). A classification algorithm for finding the optimal rank aggregation method. In *22nd international symposium on computer and information sciences*, pages 1–6.

[Arsdale and Stahovic, 2012] Arsdale, T. V. and Stahovic, T. (2012). Does neatness count? what the organization of student work says about understanding. In *Proceedings of the 119th ASEE Annual Conference and Exposition*, ASEE'12, Washington DC, United States. American Society for Engineering Education.

[Cheema et al., 2012] Cheema, S., Gulwani, S., and LaViola, J. (2012). Quickdraw: Improving drawing experience for geometric diagrams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 1037–1064, New York, NY, USA. ACM.

[Clarke et al., 2012] Clarke, A. D., Dong, X., and Chantler, M. J. (2012). Does free-sorting provide a good estimate of visual similarity? In *Proceedings of the Predicting Perceptions: The 3rd International Conference on Appearance*, pages 17–20.

[Datta et al., 2006] Datta, R., Joshi, D., Li, J., and Wang, J. (2006). Studying aesthetics in photographic images using a computational approach. In Leonardis, A., Bischof, H., and Pinz, A., editors, *Computer Vision ECCV 2006*, volume 3953 of *Lecture Notes in Computer Science*, pages 288–301. Springer Berlin Heidelberg.

[Igarashi et al., 1997] Igarashi, T., Matsuoka, S., Kawachiya, S., and Tanaka, H. (1997). Interactive beautification: A technique for rapid geometric design. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, pages 105–114, New York, NY, USA. ACM.

[Jin et al., 2003] Jin, R., Si, L., Zhai, C., and Callan, J. (2003). Collaborative filtering with decoupled models for preferences and ratings. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, pages 309–316, New York, NY, USA. ACM.

[Kim et al., 2013] Kim, H.-h., Taele, P., Valentine, S., McTigue, E., and Hammond, T. (2013). Kimchi: A sketch-based developmental skill classifier to enhance pen-driven educational interfaces for children. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*, SBIM '13, pages 33–42, New York, NY, USA. ACM.

[Li et al., 2010] Li, C., Loui, A. C., and Chen, T. (2010). Towards aesthetics: A photo quality assessment and photo selection system. In *Proceedings of the International Conference on Multimedia*, MM '10, pages 827–830, New York, NY, USA. ACM.

[Mahoney and Fromherz, 2002] Mahoney, J. V. and Fromherz, M. P. (2002). Three main concerns in sketch recognition and an approach to addressing them. In *AAAI Spring Symposium on Sketch Understanding*, pages 105–112.

[Mantiuk et al., 2012] Mantiuk, R. K., Tomaszewska, A., and Mantiuk, R. (2012). Comparison of four subjective methods for image quality assessment. *Computer Graphics Forum*, 31(8):2478–2491.

[Murugappan et al., 2009] Murugappan, S., Sellamani, S., and Ramani, K. (2009). Towards beautification of freehand sketches using suggestions. In *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling*, SBIM '09, pages 69–76, New York, NY, USA. ACM.

[ODonovan et al., 2014] ODonovan, P., Agarwala, A., and Hertzmann, A. (2014). Collaborative filtering of color aesthetics. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging (CAe)*.

[Paulson and Hammond, 2008] Paulson, B. and Hammond, T. (2008). Paleosketch: Accurate primitive sketch recognition and beautification. In *Proceedings of the 13th International Conference on Intelligent User Interfaces*, IUI '08, pages 1–10, New York, NY, USA. ACM.

[Pu and Ramani, 2007] Pu, J. and Ramani, K. (2007). Implicit geometric constraint detection in freehand sketches using relative shape histogram. In *Proceedings of the 4th Eurographics Workshop on Sketch-based Interfaces and Modeling*, SBIM '07, pages 107–113, New York, NY, USA. ACM.

[Purchase et al., 2012] Purchase, H. C., Pilcher, C., and Plimmer, B. (2012). Graph drawing aesthetics-created by users, not algorithms. *IEEE Transactions on Visualization and Computer Graphics*, 18(1):81–92.

[Rogowitz et al., 1998] Rogowitz, B. E., Frese, T., Smith, J., Bouman, C. A., and Kalin, E. (1998). Perceptual image similarity experiments. In *Human Vision and Electronic Imaging III, Proceedings of the SPIE, 3299*.

[Vatavu et al., 2011] Vatavu, R.-D., Vogel, D., Casiez, G., and Grisoni, L. (2011). Estimating the perceived difficulty of pen gestures. In *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part II*, INTERACT'11, pages 89–106, Berlin, Heidelberg. Springer-Verlag.

[Veselova and Davis, 2004] Veselova, O. and Davis, R. (2004). Perceptually based learning of shape descriptions for sketch recognition. In *Proceedings of the 19th National Conference on Artifical Intelligence*, AAAI'04, pages 482–487. AAAI Press.

[Wong and Low, 2009] Wong, L.-K. and Low, K.-L. (2009). Saliency-enhanced image aesthetics class prediction. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 997–1000.

[Wu et al., 2010] Wu, Y., Bauckhage, C., and Thurau, C. (2010). The good, the bad, and the ugly: Predicting aesthetic image labels. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 1586–1589.