

# A new approach for perceptually-based fitting strokes into straight segments

R. Plumed<sup>1</sup> P. Company<sup>2</sup> and P.A.C. Varley<sup>2</sup>

<sup>1</sup>Department of Mechanical Engineering and Construction - Universitat Jaume I, Spain

<sup>2</sup>Institute of New Imaging Technology - Universitat Jaume I, Spain

---

## Abstract

*Fitting the strokes of a sketch into geometrical primitives is still an open problem, even for sketches which depict bare line-drawings without annotations. Such sketches comprise only discrete strokes, sequences of points obtained between a pen down and a pen up.*

*It is commonly accepted that the best perceptual fittings depend on the context. Hence, we will only be able to extract the best line-drawing from a sketch by considering a complex recognition flow, where lines must be iteratively fitted according to different tentative relationships until the most plausible line-drawing is reached.*

*The recognition task considered in this paper is determining whether a stroke represents a straight line. The goal is doing it in a way that allows for iterative recognition flows. The novel contributions are that our approach is more fast and robust than accurate, uses perceptual criteria to classify strokes, and returns likeliness instead of a simple yes/no.*

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation; I.4.6 [Image processing and computer vision]: Segmentation—Edge and feature detection; J.6 [Computer Aided Engineering]: Computer Aided Design—CAD.

---

## 1. Introduction

CAD packages aimed for users ranging from beginner to competence are, to some extent, teaching the user to draw. Thus, it makes sense to start with a geometric idea of what a primitive ought to look like, and make the user draw it that way. However, outstanding designers still complain that “CAD systems still haven’t quite made that as easy as pen and paper” [Wil12]. To make full use of their conceptual design and innovation talents, outstanding designers require a sketching environment which is fully integrated with the subsequent phases of the design processes ([CCV\*09]; [JGHD09]).

Thus, we aim to develop new computer-aided conceptual design tools based on the criteria introduced by Van Dijk [VD95]. In particular, our approach is centred on sketched input. The long term goal is a Computer-Aided Sketching (CAS) package which should produce beautified line-drawings from engineering sketches, and then produce 3D models from the beautified line-drawings. But the user intent is difficult to extract since the freehand input is ambiguous in nature [MSR09]. The process to find the best perceptual

arrangement of lines of a line-drawing depends on the context [IMKT97]. Hence, we will only be able to determine such best beautified line-drawing by considering simultaneously all the mutual relationships between all the lines of the line-drawing extracted from the sketch.

We seek an approach where strokes are fitted into the most plausible types of lines, and then lines are beautified to produce the most plausible arrangement. But we cannot forget that strokes depicting polylines must be segmented before fitting, and lines fragmented into multiple strokes (dashed lines, for instance) must be grouped.

It follows that the process must be iterative (non-sequential), to compare different assumptions until a reasonable solution is reached. This means that we must run the fitting algorithms repeatedly, until the most plausible fit is reached. Hence, fast fits are more important in this context than accurate or ultra-accurate fits used in other ambits like image vectorization.

In addition, plausibility must be automatically evaluated. On the contrary of Murugappan et al. [MSR09] notwith-

standing, we do not pursue systems where different “suggestions” are obtained and the user must select the best. In fact, we do not seek the straight segment that best fits one particular stroke. Instead, we try to determine whether or not this stroke is perceived as depicting any straight segment. The particular segment depicted by the stroke is less important: it will be beautified after fitting, as fitting cannot take into account the context, while beautification can. Thus, our new approach is based on perceptual considerations, as it follows Saund and Moran’s WYPIWYG concept (What You Perceive Is What You Get) [SM94].

This paper explains the first part of our current work, where the second part is aimed at detecting candidate elliptic arcs [CPV15] and the third part is aimed at finding the most plausible beautified line-drawing that can be extracted from the input sketch. All together will constitute a fitting method, but they have been divided because of their complexity and in order to reduce the controlled variables in every study. In turn, the fitting method, together with suitable segmenting and grouping methods found in the literature, will constitute a perceptually based recognition approach.

The paper is structured as follows: Section 2 describes related work. Section 3 considers the perceptual problem. In Section 4, we describe a perceptually-based algorithm aimed at introducing likeliness in automatic perception of straight lines. Finally, in Section 5, we evaluate the approach measuring its ability to produce the same interpretation as humans. Section 6 presents our conclusions.

## 2. Related work

Since the early stages of sketch recognition [Rob63], sequential flow approaches have been predominant in sketch input. The recent approach of Wang et al. [WWGY13] is typical: stroke preprocessing precedes feature detection which precedes a hybrid-based classifier (Kara and Stahovich [KS04] and Hammond et al. [HLP\*10] are other examples). Such sequential recognition flows cannot take into account context dependencies. Only now are iterative flows being considered, one example being the loop to detect over-traced strokes in Wang et al’s FSR approach [WQG14].

Many approaches have been proposed for stroke-classification/fitting-straight-lines, ranging from the simplest, which compares chord lengths [QWJ01], to the most popular, using the Hough transform [DH72].

Some other recognition approaches have been developed or adapted specifically for Sketch-Based 3D Modelling approaches (SB3DM). Shpitalni and Lipson apply linear least squares fitting to a conic section equation [SL97]. Qin proposes a method for classifying pen strokes based on adaptive thresholds and fuzzy knowledge with respect to the linearity and convexity of the curves [Qin05]. Zhang et al. summarise older approaches, and propose a seeded segment growing algorithm for extracting graphical primitives from a

stroke [ZSZL06]. They try to refine their control parameters by using relationships between primitives. Their algorithm is reportedly reliable for detecting straight segments. A variant of the Hough transform for SBM was recently introduced by Plumed et al. [PCV13].

The first problem is that all those approaches assume that the stroke fits into a straight line. Hence, the only possible acceptable outputs are the best geometrical fit or a failure condition. In our approach, we ask for a fit before we know if this fit is viable and desirable. Thus, we seek likeliness, instead of deterministic evaluations (*yes/no*) that do not allow for collaborative-decision recognition flows.

The second problem is that filters used in SBM approaches are specifically aimed at removing sketching noise (which is usually much bigger and less isolated than optical or electronic noises, common while processing camera images). In addition, to the best of our knowledge, thresholds used in this literature were estimated by the authors without taking into account how well they correlate with human perception. The excellent approach suggested by Shpitalni and Lipson in 1997 is still representative: a minor to major axis ratio 1:20 is used to distinguish strokes depicting elliptic arcs from line segments. This ratio is heuristic and was based only on the authors’ observations. This example illustrates how arbitrary choice of thresholds governs the behaviour of existing algorithms. Our claim is that those thresholds should be re-evaluated to agree with human perception.

Finally, two considerations are needed to better fix the scope of this work. Firstly, while building sketch understanding systems, an important distinction must be done between systems aimed at producing 2D formal and structured representations, from those aimed at producing 3D models. In SB3DM, the beautified line-drawing is just an intermediate output. For instance, Murugappan et al.’s paper [MSR09] is very 2D-specific (one sole image depicts a 3D object, and the implicit geometric constraints inferred are limited to endpoints coincidence), and the reliance on 2D constraint solving means that it can be hardly extended to interpreting 2D drawings of 3D objects, as it beautifies the drawing in excess (which may prevent using the drawing to produce the 3D model). Secondly, our approach deals with what Onkar and Sen [OS10] call *defining-strokes*, containing minimal information: only coordinates of points plus time elapsed from previous point. This input would be classified by Chansri and Koomsap [CK12] as “Non-overtraced online freehand sketches”.

## 3. Perceptually-based fitting line-segments

Our intention, when classifying strokes, is that very good strokes are identified as *straight*, very bad strokes are identified as *not straight*, and in between there are doubtful cases to which a merit figure is assigned.

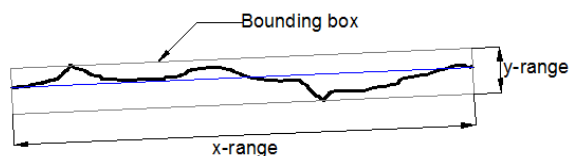
To this end, our approach uses the *Tolerance*, which is a well-known concept in Geometric Dimension and Tolerancing for measuring the “straightness” of a line (ISO 1101-1983). Given the bounding box of the line and defining *x-range* and *y-range* as in Figure 1, the absolute tolerance of straightness is:

$$Tolerance = |y - range| \quad (1)$$

The lower this parameter is, the straighter the stroke is considered to be.

One important concept in manufacturing is surface finish, where the nature of a surface is defined by the 3 characteristics: lay, surface roughness, and waviness. They define the small local deviations of a surface from the perfectly flat ideal. When representing 3D surfaces in 2D, we lose the lay parameter which determines the main direction of the surface texture and it is related to the machine tool used. Surface “irregularities” are then determined by long wavelength shapes or “waviness” and short wavelength features or “roughness”, so that waviness is a kind of carrier wave and roughness is the modulation over it. Its measurement and control are part of a frame work of Geometrical Product Specifications (GPS) standards (like ISO 4287:1997). But, since we are not considering imperfections of actual shapes, we have replaced the term waviness by undulation and roughness by oscillation, which are terms commonly used to measure noise in signal processing. To sum up, in this paper, undulations are low frequency changes in the direction of the stroke around the theoretical straight line while oscillations are high frequency changes.

At this respect, we note that *Tolerance* parameter does not distinguish whether the lack of straightness results from oscillations or undulations.



**Figure 1:** Representation of the bounding box and its dimensional parameters.

To determine the limits of acceptance of humans, we need to ask humans in experiments asking individuals from representative populations. We have designed and implemented questionnaires to identify which strokes humans perceive as depicting straight lines, and which they consider cannot be straight lines. The difficult point here is that experiments must identify psychological behaviour, without accidentally misinterpreting learned behaviour of particular subjects as general perceptual behaviour. To ensure the efficiency of the experimental analysis we fulfill the requirements proposed

by Simmons et al. [SNS11] when designing experiments. One of the common rules to finish the data collection is to collect valid responses of at least 20 subjects.

We briefly describe the two experiments we performed. The intention of our experiments was to validate or reject the following hypotheses:

1. *Tolerance* matches human perception of relative straightness of lines.
2. Humans are more prone to interpret undulations as intended alternative shapes, while they are more prone to interpret oscillations as involuntary errors.

### 3.1. Experiment #1 to mimic human perception

We conducted an experiment comparing the relative straightness of a new set of twelve strokes (Figure 2), of similar length and orientation but with increased values of *Tolerance*.

#	Stroke	Tolerance
1		0,56
2		2,16
3		2,63
4		2,69
5		4,30
6		4,38
7		6,19
8		6,48
9		7,62
10		7,99
11		11,07
12		11,80

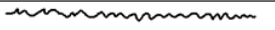
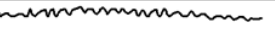
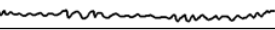
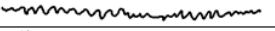
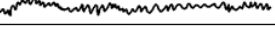
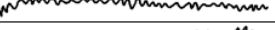
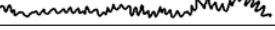
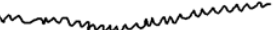
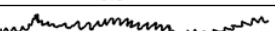

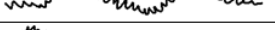
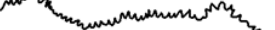
**Figure 2:** Example strokes for the relative straightness test.

We interviewed a sample of 22 subjects (mainly engineering teachers and students, plus a few from other backgrounds). They were given the set of twelve strokes on separate A6 sheets which had been shuffled to randomise, and were asked to re-order the sheets in order of decreasing straightness. No differences due to subject backgrounds were detected in the results.

We conclude that *Tolerance* usually matches human perception. 18 subjects out of the 22 classified Strokes 1, 2, 3 and 4 as good; Strokes 6 and 7 as average, and Strokes 10, 11 and 12 as poor. This result is statistically significant ( $p(X \geq 18) = 0.0021$ , in the binomial distribution  $B(22, 1/2)$ , and assuming an alpha level of  $\alpha = 0.01$ ). In a complementary study [PCV14], we concluded that discrepancies are explained because humans understand that corners break straightness more than undulations do.

### 3.2. Experiment #2 to mimic human perception

To validate the hypothesis that humans generally distinguish oscillations from undulations, and are more prone to interpret undulations as intended alternative shapes, while they are more prone to interpret oscillations as involuntary errors, we defined a new set of strokes, similar to the previous group but including oscillations, as shown in Figure 3.

#	Stroke	Tolerance
-1		3.58
-2		4.24
-3		4.19
-4		4.59
-5		6.24
-6		7.53
-7		8.48
-8		8.36
-9		8.97
-10		9.07
-11		12.63
-12		12.58

**Figure 3:** Example strokes with oscillations for the relative straightness test.

Then we asked another 22 subjects (again, mainly engineering teachers and students, with a few from other backgrounds) to order A6 sheets containing the twelve oscillating strokes in order of decreasing straightness. Results detailed in a complementary study [PCV14] show that oscillating strokes are ordered by humans in the same way as non-oscillating ones. This result is statistically significant ( $p(X \geq 16/22) = 0.021$ ), with a significance of  $\alpha = 0.05$  and confirms that oscillations are perceived separately from undulations, as when all the compared strokes contain the same sort of oscillations, humans filter out the oscillations and classify the strokes using corners and undulations.

In that study [PCV14], we also observed that some subjects mentally filter out oscillations and evaluate the smoothed stroke with a penalty, reducing the mark from *good* to *average* or from *average* to *poor*.

### 3.3. Experiment #3 to mimic human perception

Based on results of Experiment #2, we deduce that a significant part of the population seems to mentally filter oscillations and perceive the straightness of the underlying stroke. To validate or reject this new hypothesis, we randomly mixed two oscillating with ten non-oscillating strokes, without repeating any type in the same set.

We interviewed 32 new subjects, divided into two groups.

In the first group, we performed sixteen different tests using four strokes of types 1, 6, 7 and 11 (as they had proved to be more stable in their perception), and only two randomly-chosen strokes of the 12 oscillating types. In the second group, we asked the other 16 subjects (mainly engineering teachers and students, with a few from other backgrounds) to order these mixed sets. In this case, the sets comprised ten non-oscillating strokes plus two oscillating strokes, where the oscillating strokes were selected only from the good and average subsets (excluding Strokes -9 to -12).

The first results showed that between 30 and 50% of the subjects listed the two oscillating strokes as the least straight.

Results also show that everybody distinguishes oscillations from undulations, but in two different ways. Some interviewed subjects discarded oscillating strokes because of their poor quality as geometric straight lines. Other subjects mentally filtered out oscillations and evaluated the smoothed stroke. Our observation was reinforced by the queries of some subjects asking whether they had to pay attention or ignore the lack of “smoothness”, “flatness” or “horizontality” of some lines (their queries were not answered). Analysing in more detail the group of subjects who spontaneously smoothed the oscillating strokes, we found that they did not consider oscillating strokes to be quite as straight as their smoothed equivalents; oscillating strokes are devalued to some extent. There are even cases where all oscillating strokes, good and bad alike, were placed together as an intermediate category, worse than good non-oscillating strokes but better than bad non-oscillating strokes—instead of reducing the judged quality of the stroke, the oscillations reduced the subjects’ ability to judge the stroke.

To investigate this further, we modified the experiment, including only oscillating strokes of the first two groups (-1 to -8) as we assume that oscillating strokes of the last group (-9 to -12) would not be assessed ahead of their non-oscillating equivalent (and, obviously, cannot be placed behind their equivalents). As we suspected that subjects with knowledge of engineering concepts (such as signal/noise or surface imperfections) may be more prone to filter out oscillations, we interviewed only subjects with engineering backgrounds (so our conclusions would not be valid to describe human perception in general, but we are interested here in how engineers perceive sketches). Since some subjects returned segregated sets of strokes, more subjects were interviewed until we had 16 non-segregated classifications.

Since no simple and systematic penalty can be observed (perhaps the sample is too small to quantify it), we cannot validate the fifth hypothesis. However, we still can conclude that the devaluation exists, and many of the subjects (at least 50%, and more in the case of subjects with engineering backgrounds) tend to evaluate the smoothed line for straightness and then reduce the mark from good to average or from average to poor.

#### 4. Perceptually-based algorithm for fitting line-segments

As described in Section 2, the common approaches for fitting a straight line with unknown parameters to a set of stroke points are geometrical: based on minimising the total error between the stroke and the fitted line. They are time-consuming and add a geometrical precision which is unnecessary for interpretation of sketches. Instead, we are interested in determining whether the stroke depicts a straight line, not which line it depicts. Thus, the new approach works in two stages: calculate a reasonably good and very fast fit, and calculate how well the stroke fits into a narrow tolerance band around the fitted line.

Our approach starts by finding the straight line which connects the two endpoints of the stroke. This is the most obvious *perceptual* fit we could use, since endpoints coincidence is the simplest and quite usual way to interact with other neighbourhood strokes [IMKT97]. In fact, other geometric constraints (horizontal or vertical alignment, parallelism, perpendicularity, etc.) can hardly be introduced at this stage, as they clearly depend on the context, which is yet undefined. Although we are currently analysing isolated strokes, we must remember that they are assumed to belong to a sketch. Hence, in the pursued full recognition approach, perceptual relationships with neighbour strokes will be more important than geometrical straightness, position or orientation. For example, in Figure 4, people perceive Stroke 0 as connected to Strokes 1 and 3, and parallel to Stroke 2, regardless of the position and orientation of any geometrical fit, and even regardless of the quality of the sketch (*good* in Figure 4 left, *poor* in Figure 4 right).

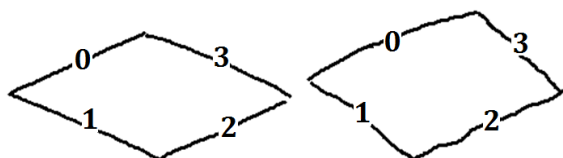


Figure 4: Examples of mutual relationships between strokes.

From the line which connects the two endpoints, we determine whether the stroke depicts a straight line, not which line it depicts. To this end, we chose the *Tolerance* metric. Finally, as we do not need a hard *yes/no* classification, we compare the tolerance of the stroke with two thresholds: minimum and maximum tolerance (respectively *TolMin* and *TolMax*). For tolerances between these two values, this returns a figure of merit between 0.0 (*no*) and 1.0 (*yes*).

The main stages of the algorithm are as follows:

1. Define the segment as the straight line connecting both endpoints of the stroke.
2. Calculate the slope of the line, and rotate the stroke.
3. Calculate *Tolerance* for the original stroke.
4. If *Tolerance* is below the minimum tolerance, accept the stroke as a straight line with a figure of merit of 1.0 (certain), and end the algorithm.
5. Smooth the stroke (to prevent oscillations from affecting how it is to be evaluated as a straight line), and store the number of smoothing stages (*NSS*).
6. Calculate *Tolerance* for the *smoothed* stroke.
7. Assign a merit figure of 1 for *Tolerance* below *TolMin*, 0 for *Tolerance* above *TolMax*, and linearly decreasing from 1 to 0 within this range (so as to distinguish *good*, *average* and *poor* strokes).
8. Reduce the figure of merit for oscillating strokes, depending on a devaluation parameter (*Penalty*) and the number of smoothing stages (*NSS*):  $Merit = Merit - Penalty * NSS$ . Negative merits are not allowed.

Stroke numbers	Minimum and maximum tolerance bands
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	

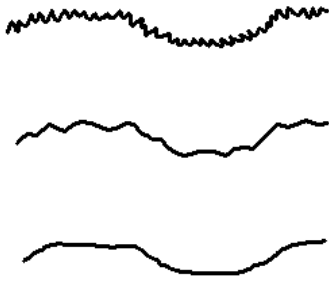
Figure 5: Minimum and maximum tolerance bands for the twelve strokes of Figure 2.

The orthogonal bounding box of the rotated stroke gives good initial estimates of the stroke's length and width. Such geometrically imprecise metrics do not conflict with our perceptual approach since they are not deterministic and this is an early step in a complex flow where dubious decisions will be revisited in the light of redundant, contradictory or merely related cues.

#### 4.1. Smoothing

There are well-known techniques for removing high frequency oscillations [Vas08], of which moving average (or rolling average) is perhaps the most popular. However, it is also well-known that estimating the right parameters for

distinguishing noise from signal is critical [Dur59]. Since we are interested in what humans perceive, we opted for a very simple method which can be easily controlled by those parameters which seem to be the most important for humans: corners and width. Thus, our approach removes “micro-corners”, taking advantage of the fact that strokes are sequences of points (not clouds of points) to sequentially remove alternate points of the stroke as long as the smoothed stroke still contains corners, and while the width of the resulting stroke is still similar to the width of the original stroke. The metric for *similarity* is the maximum permitted variation in the stroke width while smoothing it, defined as threshold *TolSmooth*—we only remove alternate points whenever their mutual distance is lower than  $2 * TolSmooth$ . The number of smoothing steps (*NSS*) required for every stroke is recorded. The effect of smoothing can be seen in Figure 6.



**Figure 6:** Oscillating and undulating stroke (above), after four smoothing steps (centre), and similar undulating but non-oscillating stroke (down).

The approach uses one of three segmentation methods to find corners: IStraw (as implemented by Xiong and LaViola [XL10]), Shortstraw [WEH08] and Sliding strips (proposed by Masood and Sarfraz [MS07]). The initial choice is made by the user, with IStraw as the default. Since IStraw requires timing information, if it is selected but no timing information is available, Shortstraw is used instead.

Note that, although corners are calculated, strokes are not actually segmented, since we attempt to fit the line *before* deciding whether the stroke must be segmented. Here, we only use segmentation information at that end.

## 5. Analysis

In this section we first determine the best tuning parameters for our approach. We then evaluate the ability of the approach to produce the same interpretation as humans. Next, we evaluate running time, showing that the approach improves over NHT [PCV13] for all reasonable input and is fast and reliable enough for an on-line Sketch-Based Modelling application. A more detailed analysis of the algorithm

shows that it generally behaves as humans do, but that it is sensitive to variations in two of its four parameters and that it is still unable to solve some pathological cases.

### 5.1. Performance of the algorithm for fitting line segments

Applying the same set of examples to our algorithm, our algorithm was found to replicate human behaviour for the following recommended parameters: *TolMin* 3.5% of the stroke’s length (measured as the bounding box length), *TolMax* 7.0% of the stroke’s length, *TolSmooth* 1.0% of the stroke’s length, and *Penalty* 10% for every smoothing step. These values are validated with the sensitivity analysis detailed in section 5.4. The recommended values are in the proper range.

Table 1 shows the figures of merit returned by the algorithm and the number of smoothing steps required for each stroke.

**Table 1:** Figures of merit returned by the algorithm

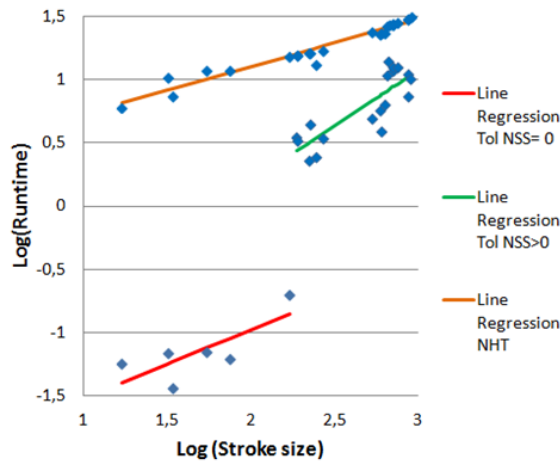
Stroke	Merit	NSS	Stroke	Merit	NSS
1	1.0	0	-1	0.50	5
2	1.0	0	-2	0.50	5
3	1.0	0	-3	0.37	5
4	1.0	0	-4	0.50	5
5	0.47	3	-5	0.21	6
6	0.94	0	-6	0.10	5
7	0.0	4	-7	0.0	6
8	0.18	0	-8	0.0	5
9	0.0	3	-9	0.0	6
10	0.0	4	-10	0.0	6
11	0.0	4	-11	0.0	5
12	0.0	3	-12	0.0	6

From Table 1, strokes with a figure of merit equal or greater than 0.5 correspond to strokes perceived as acceptable by humans, and this criterion could be used in situations where an immediate *yes/no* decision is preferred.

### 5.2. Running time

Figure 7 illustrates the practical time complexity of the Tolerance and NHT algorithms (run time is in milliseconds and size is the number of points in the stroke). The critical stage of the Tolerance algorithm is the smoothing oscillation process, but this stage is only executed when the stroke does not reach the highest possible figure of merit (1.0), thus the algorithm has a different time cost depending on whether the stopping criterion of step 3 is met or not: it has a fast response,  $O(n^{0.54})$ , when  $NSS=0$  (red line regression), and a slower one,  $O(n^{0.88})$ , when  $NSS > 0$  (green line regression). The NHT algorithm (orange line), although having a lower time complexity,  $O(n^{0.38})$ , has a much higher time constant and is, as expected, slower overall.

We can thus conclude that our algorithm is fast enough for use in an on-line SBM application.



**Figure 7:** Run time comparison between Tolerance and NHT algorithms.

**5.3. General performance**

To compare the results of the Tolerance algorithm with human perception, first we calculated how often the subjects classified each stroke as good, average or poor, using the results of Experiments #1 and #2, where strokes with and without oscillations were independently classified, and of Experiment #3 (explained in detail in the complementary study [PCV14]), where both types of stroke are randomly combined. As Table 2 shows, figures of merit from the Tolerance algorithm fit well with frequencies of Experiment #3 if we apply the following criteria: the stroke is classified as good when its figure of merit is  $\geq 0.5$ ; as poor when its figure of merit is zero, and average in other case.

**Table 2:** Perception frequencies of experiments and figures of merit returned by the algorithm

Stroke	Experiments #1 and #2			Experiment #3			Algorithm Merit
	Good	Average	Poor	Good	Average	Poor	
1	100%			100%			1
2	100%			100%			1
3	100%			100%			1
4	100%			96.2%	3.8%		1
5		77.3%	22.7%	14.3%	81.0%	4.8%	0.47
6		100%	0%	71.4%	28.6%		0.94
7		90.9%	9.1%		81.3%	18.8%	0
8		90.9%	9.1%	10%	80%	10%	0.18
9		18.2%	81.8%		45%	55%	0
10		9.1%	90.9%		35%	65%	0
11			100%			100%	0
12		13.6%	86.4%		15%	85%	0
-1	100%			57.1%	28.6%	14.3%	0.5
-2	100%			40%	40%	20%	0.5
-3	90.9%	9.1%		25%	75%	75%	0.37
-4	100%			25%	75%		0.5
-5		100%			100%		0.21
-6	9.1%	90.9%			100%		0.1
-7		72.7%	27.3%		16.7%	83.3%	0
-8		54.5%	45.5%			100%	0
-9		72.7%	27.3%			100%	0
-10			100%			100%	0
-11			100%			100%	0
-12			100%			100%	0

Comparing the results of the experiments with the merits calculated by the algorithm, we can see that, in general, the algorithm replicates human judgement, but does not fit correctly the perceptual results for Strokes 7, -3 and -4 in Experiment #3. These cases are analysed in more depth in Section 5.5.

**5.4. Sensitivity analysis**

We performed a sensitivity analysis to determine the robustness of the Tolerance algorithm to variations in the tuning parameters. The algorithm uses the four parameters described in Section 4 (*TolMin*, *TolMax*, *TolSmooth*, *Penalty*) and tuned as described in Section 5.1. We varied all parameters in steps around their recommended values to find the ranges within which the parameters still work reasonably well (Table 3), and determined how many more mistakes the algorithm makes.

We conclude that the algorithm is sensitive to *TolMin* and *TolMax*, and the mistakes increase as soon as the parameters vary slightly from their recommended values. The algorithm is robust to varying *TolSmooth* and *Penalty* as they do not influence those cases where *NSS=0* (Table 3), or when the merit is assigned directly (1 for *Tolcob* < *TolMin* or 0 for *Tolcob* > *TolMax*). However, we shall see their influence in more detail in the next section.

**Table 3:** Parameters of the algorithm and their recommended range of values

	Tested		Recommended						Tested	
	Min	Errors	Min	Errors	Default	Errors	Max	Errors	Max	Errors
<i>TolMin</i>	0.01	6/24	2.5	5/24	3.5	3/24	5	4/24	15	17/24
<i>TolMax</i>	3	9/24	6	5/24	7	3/24	11	4/24	60	15/24
<i>TolSm</i>	0.1	7/24	0.5	4/24	1	3/24	2	2/24	20	4/24
<i>Penalty</i>	0	5/24	7	3/24	10	3/24	12	6/24	30	7/24

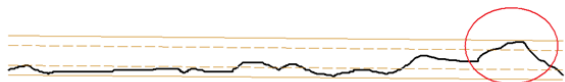
When *TolMin* equals *TolMax*, strokes cannot be classified as average and the figures of merits only have two possible values: 0 or 1.

**5.5. Pathological cases**

As mentioned above, the algorithm does not correctly fit human perception for Strokes 7, -3 and -4 in Experiment #3 (Table 2). Here we study these examples in detail, as they are representative of the most common pathological cases found during the experiments.

Stroke 7 has a sharp undulation close to its right end point (Figure 8). This isolated undulation affects the location of the tolerance bands, resulting in a low figure of merit. Further, the smoothing *Penalty* (the Stroke 7 requires 4 smoothing steps) reduces the merit to zero. However humans seem to filter out this isolated undulation, as Table 2 shows for both Experiments #1 and #3 that humans usually classified Stroke 7 as average. Reducing *Penalty* (from 10% to

7%) results in a figure of merit which classifies the stroke according to human perception, but this smaller Penalty is not a good general value, since other strokes (5 and -3 in the example) would then be classified incorrectly.



**Figure 8:** Stroke 7 and the tolerance bands (according to *TolMin* and *TolMax* parameters).

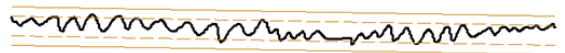
Stroke -3 (Figure 9) is a particular case where human perception seems anomalous. Subjects tend to classify the oscillating strokes used in Experiments #2 and #3 as (1) belonging to the same group as their equivalent non-oscillating strokes, or (2) moved to the next lower group. However in this case, most subjects classified Stroke -3 as good in Experiment #2, while they mainly classified it as poor in Experiment #3. Notwithstanding this anomaly of human perception, the algorithm still behaves as humans do.



**Figure 9:** Stroke -3 and the tolerance bands (according to *TolMin* and *TolMax* parameters).

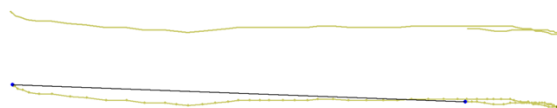
In Experiment #2, all polled subjects classified Stroke -4 (Figure 10) as good; in Experiment #3, 75% of subjects classified Stroke -4 as average, while 25% still classified it as good. Since the underlying stroke (after smoothing out oscillations) has no noticeable undulations, those subjects who perceive the underlying shape tend to classify this stroke as *good*, while those who perceive oscillations as imperfections are liable to classify it as *average* or even *poor*.

The algorithm behaves in a similar way: The smoothed stroke is good (it fits within the minimum tolerance band) but devaluation may make it appear as average, as this example is very sensitive to Penalty—increasing Penalty from 10% to 11% would switch the stroke from good to average.



**Figure 10:** Stroke -4 and the tolerance bands (according to *TolMin* and *TolMax* parameters).

Finally, the approach has proved unable to process self-intersecting strokes (Figure 11) correctly: the approach returns a shorter line than the psychologically expected one. We intend to solve this drawback in future versions by adding a check to detect cases where the stroke bounding box is longer than the segment length.



**Figure 11:** Self-intersecting stroke, and its fitted segment.

## 6. Conclusions

Humans produce sketches which are inherently imperfect from the point of view of geometry, but human perception includes the ability to filter out imperfections and detect the underlying geometry. Current recognition approaches are biased towards geometrical accuracy or ultra-accuracy, and pay insufficient attention to human perception. We argue that algorithms should accept what humans accept, should reject what humans reject, and should doubt where humans doubt.

The contribution here is twofold. Firstly, we propose a new algorithm which matches human interpretation of straight segments acceptably well and is fast enough for use in online applications. Secondly, we report the results of experiments to determine which strokes humans perceive as depicting straight lines, and which they consider cannot be straight lines.

## Acknowledgments

This work was partially funded by financial support from the Ramon y Cajal Scholarship Programme and by the “Pla de Promoció de la Investigació de la Universitat Jaume I”, project P1 1B2010-01. We wish to thank Salvador Mondragón, who collected many questionnaires from his students, and Margarita Vergara, for her contribution to statistical data treatment.

## References

- [CCV\*09] COMPANY P., CONTERO M., VARLEY P., ALEIXOS N., NAYA F.: Computer-aided sketching as a tool to promote innovation in the new product development process. *Computers in Industry* 60(8) (2009), 592–603. doi:10.1016/j.compind.2009.05.018. 1
- [CK12] CHANSRI N., KOOMSAP P.: Automatic single-line drawing creation from a paper-based overtraced freehand sketch. *International Journal of Advanced Manufacturing Technology* 59(1-4) (2012), 221–42. doi:10.1007/s00170-011-3487-z. 2
- [CPV15] COMPANY P., PLUMED R., VARLEY P.: A fast approach for perceptually-based fitting strokes into elliptical arcs. *The Visual Computer* (2015). doi:10.1007/s00371-015-1099-6. 2
- [DH72] DUDA R., HART P.: Use of the hough transformation to detect lines and curves in pictures. *Communication of the ACM* 15(1) (1972), 11–15. doi:10.1145/361237.361242. 2
- [Dur59] DURBIN J.: Efficient estimation of parameters in moving average models. *Biometrika* 46 (1959), 306–16. doi:10.1093/biomet/46.3-4.306. 6



- [HLP\*10] HAMMOND T., LOGSDON D., PAULSON B., JOHNSTON J., PESCHEL J., WOLIN A., T. T.: A sketch recognition system for recognizing free-hand course of action diagrams. *Proceedings of the Twenty-Second Innovative Applications of Artificial Intelligence Conference (IAAI-10)* (2010), 1781–86. [doi:10.1145/263407.263525](https://doi.org/10.1145/263407.263525). 1, 5
- [IMKT97] IGARASHI T., MATSUOKA S., KAWACHIYA S., TANAKA H.: Interactive beautification: A technique for rapid geometric design. in *Proceedings of UIST '97* (1997), 105–14. [doi:10.1145/263407.263525](https://doi.org/10.1145/263407.263525). 1, 5
- [JGHD09] JOHNSON G., GROSS M., HONG J., DO E.: Computational support for sketching in design: A review. *Foundations and Trends in Human-Computer Interaction 2(1)* (2009), 1–93. [doi:10.1561/1100000013](https://doi.org/10.1561/1100000013). 1
- [KS04] KARA L., STAHOVICH T.: Hierarchical parsing and recognition of hand-sketched diagrams. *UIST: Proceedings of the Annual ACM Symposium on User Interface Software and Technology* (2004), 13–22. [doi:10.1145/1029632.1029636](https://doi.org/10.1145/1029632.1029636). 2
- [MS07] MASOOD A., SARFRAZ M.: Corner detection by sliding rectangles along planar curves. *Computers and Graphics 31* (2007), 440–48. [doi:10.1016/j.cag.2007.01.021](https://doi.org/10.1016/j.cag.2007.01.021). 6
- [MSR09] MURUGAPPAN S., SELLAMANI S., RAMANI K.: Towards beautification of freehand sketches using suggestions. In *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling, SBIMDesign Studies* (2009), 69–76. [doi:10.1145/1572741.1572754](https://doi.org/10.1145/1572741.1572754). 1, 2
- [OS10] ONKAR P., SEN D.: Functional segmentation of strokes for product sketch understanding. *International Journal of Shape Modeling 16(1 and 2)* (2010), 9–38. [doi:10.1142/S0218654310001262](https://doi.org/10.1142/S0218654310001262). 2
- [PCV13] PLUMED R., COMPANY P., VARLEY P.: Human-like recognition of straight lines in sketched strokes. *CEIG 2013, Spanish Computer Graphics Conference* (2013), 56–66. 2, 6
- [PCV14] PLUMED R., COMPANY P., VARLEY P.: Study on perceptually-based fitting line-segments. *Technical Report Ref. 08/2014, Geometric Reconstruction Group*, <http://www.regeo.uji.es>, Regeo (2014). 3, 4, 7
- [Qin05] QIN S.: Intelligent classification of sketch strokes. *IEEE EUROCON2005-Computer as a Tool* (2005), 1374–77. [doi:10.1109/EURCON.2005.1630216](https://doi.org/10.1109/EURCON.2005.1630216). 2
- [QWJ01] QIN S., WRIGHT D., JORDANOV I.: On-line segmentation of freehand sketches by knowledge-based nonlinear thresholding operations. *Pattern Recognition 34* (2001), 1885–93. [doi:10.1016/S0031-3203\(00\)00122-9](https://doi.org/10.1016/S0031-3203(00)00122-9). 2
- [Rob63] ROBERTS L.: *Machine perception of three-dimensional solids*. Lexington, Mass: M.I.T. Technical report 315, Lincoln Laboratory, 1963. 2
- [SL97] SHPITALNI M., LIPSON H.: Classification of sketch strokes and corner detection using conic sections and adaptive clustering. *Journal of Mechanical Design, Transactions of the ASME 119(1)* (1997), 131–5. [doi:10.1115/1.2828775](https://doi.org/10.1115/1.2828775). 2
- [SM94] SAUND E., MORAN T.: A perceptually-supported sketch editor. *Proceedings UIST* (1994), 175–84. [doi:10.1145/192426.192494](https://doi.org/10.1145/192426.192494). 2
- [SNS11] SIMMONS J., NELSON L., SIMONSOHN U.: False-positive psychology: undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological science 22* (2011), 1359–66. [doi:10.1177/0956797611417632](https://doi.org/10.1177/0956797611417632). 3
- [Vas08] VASEGHI S.: *Advanced Digital Signal Processing and Noise Reduction*, 4rd ed. Wiley, 2008. [doi:10.1002/0470094966](https://doi.org/10.1002/0470094966). 5
- [VD95] VAN DIJK C.: New insights in computer-aided conceptual design. *Design Studies 16(1)* (1995), 62–80. [doi:10.1016/0142-694X\(95\)90647-X](https://doi.org/10.1016/0142-694X(95)90647-X). 1
- [WEH08] WOLIN A., EOFF B., HAMMOND T.: Shortstraw: a simple and effective corner finder for polylines. *SBIM 2008* (2008), 33–40. [doi:10.2312/SBM/SBM08/033-040](https://doi.org/10.2312/SBM/SBM08/033-040). 6
- [Wil12] WILLIS A.: Adrian newey: F1 design that starts with a pad and pencil. humans invent. *Website: http://www.humansinvent.com/!8011/adrian-newey-f1-design-that-starts-with-a-pad-and-pencil/* (2012), Last accessed: September1ts. 1
- [WQG14] WANG S., QIN S., GAO M.: New grouping and fitting methods for interactive overtraced sketches. *Visual Computer 30(3)* (2014), 285–297. [doi:10.1007/s00371-013-0844-y](https://doi.org/10.1007/s00371-013-0844-y). 2
- [WWGY13] WANG S., WANG G., GAO M., YU S.: Using fuzzy hybrid features to classify strokes in interactive sketches. *Advances in Mechanical Engineering 2013, Article ID 259152* (2013), 7 pages. [doi:10.1155/2013/259152](https://doi.org/10.1155/2013/259152). 2
- [XL10] XIONG Y., LAVIOLA J.: A shortstraw-based algorithm for corner finding in sketch-based interfaces. *Computers and Graphics 34(5)* (2010), 513–27. [doi:10.1016/j.cag.2010.06.008](https://doi.org/10.1016/j.cag.2010.06.008). 6
- [ZSZL06] ZHANG X. W., SONG J. Q., Z. D. G., LYU M. R.: Extraction of line segments and circular arcs from freehand strokes based on segmental homogeneity features. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 36(2)* (2006), 300–11. [doi:10.1109/TSMCB.2005.857288](https://doi.org/10.1109/TSMCB.2005.857288). 2