FYP Presentation

Shape Processing and its Application to Stroke Rendering and Stylization

"An automated inking system"

By

Tsang Hao Fung

Advised by

Prof. Pedro SANDER

Abstract

- Goal: generates ink work from plain curves
- Address existing limitations
- Describe a stack of algorithms
- Present a data-driven stylization system
- Demonstrate a working system

A trailer

In this image, one column was computer generated, while the other was scanned. Can you guess?

Scope of study

Solid ink only. No color, no texture

Artwork courtesy: Alberto Lung (left), CNArtGallery (right)

http://robotninjamonsters.blogspot.hk/2010_06_01_archive.html http://www.artisoo.com/blog/how-to-use-a-chinese-calligraphy-brush/

Expressiveness of brushes

Artwork courtesy: Alberto Lung (left), 李克杰 (right)

The unfortunate flatness of digital strokes

Left: ***Toy Tinkers***, 1949. Right: ***Rescue Rangers***, 1990.

Walt Disney Productions.

Related work

Curvature-based stroke rendering [S. Saito et al. 2008]

S. Saito, A. Kani, Y. Chang, and M. Nakajima, "Curvature-based stroke rendering", The visual computer, 24(1), 1-11, 2008.

Related work

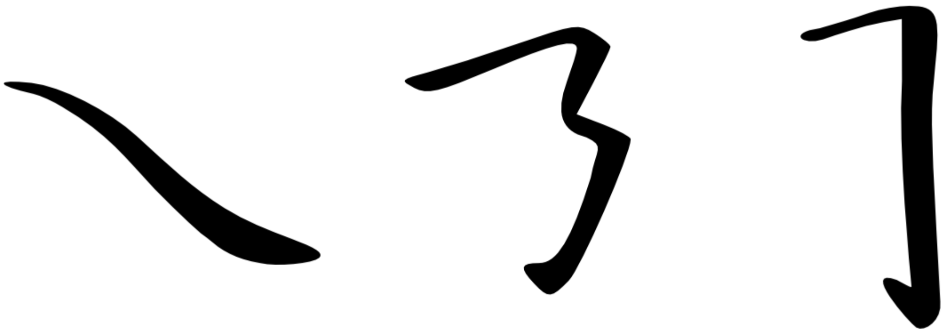
Data driven paint synthesis [Jingwan Lu et al. 2013]

Left: stroke library. Right: stylized result.

J. Lu and C. Barnes and S. DiVerdi and A. Finkelstein, "RealBrush: Painting with Examples of Physical Media", ACM Transactions on Graphics, 32(4), 117, 2013.

Their problems

- Piecewise stylization
    - Do not consider stroke interactions
- Do not support expressive brush strokes
    - Inherit limitation of the stroke model

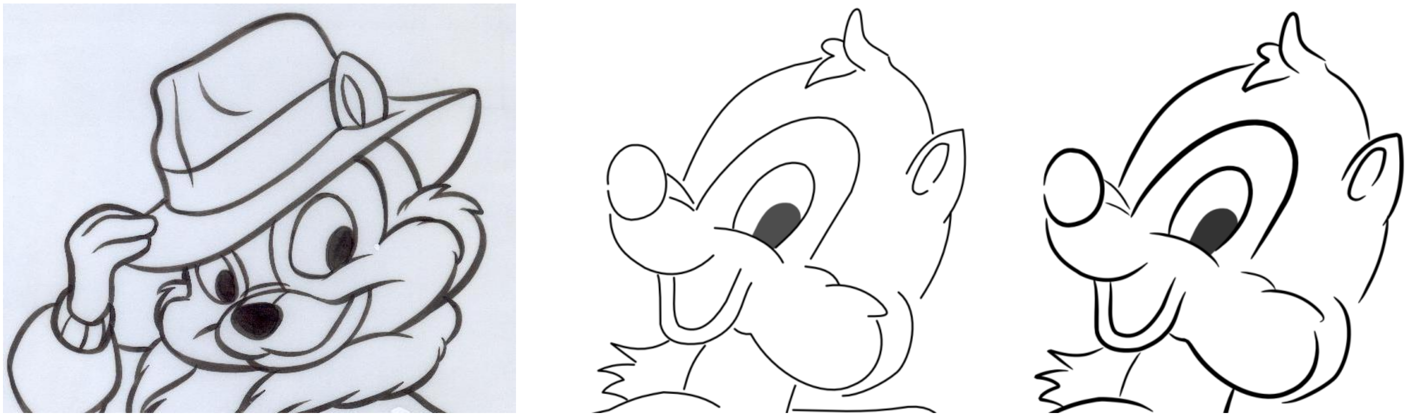


Impossible strokes

See P.21

Problem statement

Suppose some scans of calligraphy or inked illustrations drawn by artists are obtained. Given some unstroked input curves, the system needs to apply strokes to the sketched curves in a style that mimics the reference style.



Left: reference. Center: input. Right: example output.

http://animationasart.rubberslug.com/gallery/inv_info.asp?ItemID=353533

Generative stroke model

It models the stroke as the resulting shape swept by moving a brush along a trajectory.

Convolution / Minkowski Sum

The traditional definition of a stroke S is defined in terms of the convolution of a trajectory T with a fixed brush B.

$$S = T * B$$

where $*$ is defined as

$$A * B = \cup \{a + b \mid a \in A, b \in B\}$$

But B is constant! Can we parameterize them?

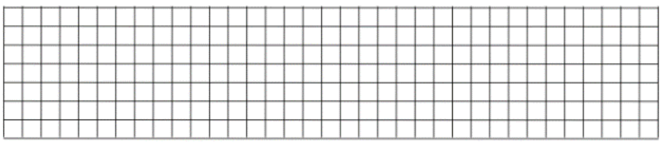
$$S = T(t) * B(t) = \cup_t \{a + b \mid a = T(t), b \in B(t)\}$$

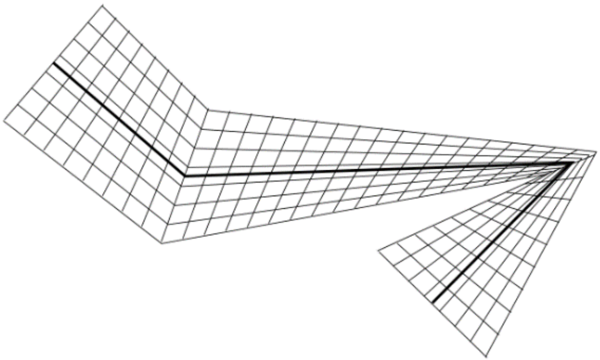
Yes, but it is extremely hard to process shapes under this theoretic framework.

Non generative stroke model

Skeletal strokes

- a texture is deformed by a skeleton to form a stroke
- preserves texture well, but distorts shape seriously





Left: figure. Right: skeletal stroke.

The grand problem

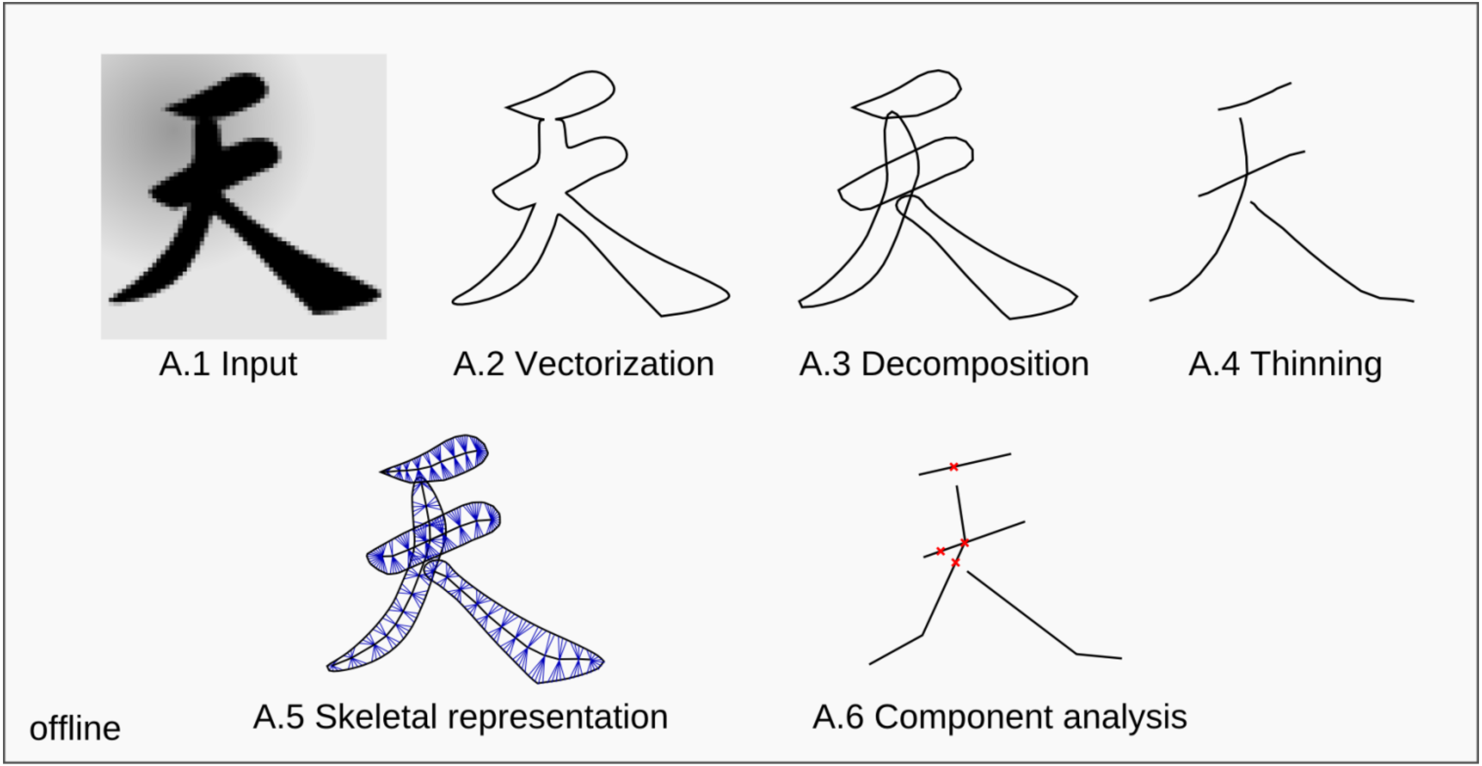
- stylization is constrained by the underlying stroke model
- stroke model is defined by the underlying shape theory

Our approach

- contour based shape processing
- vector pipeline
- non generative stroke model

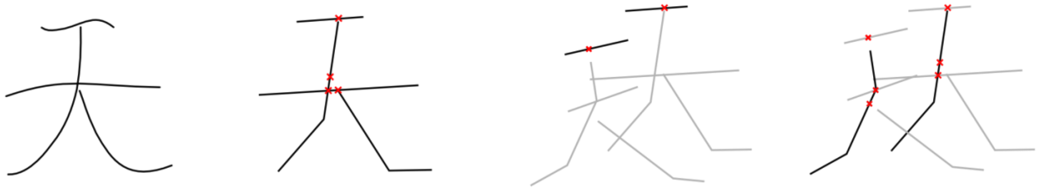
The automated inking system

Offline

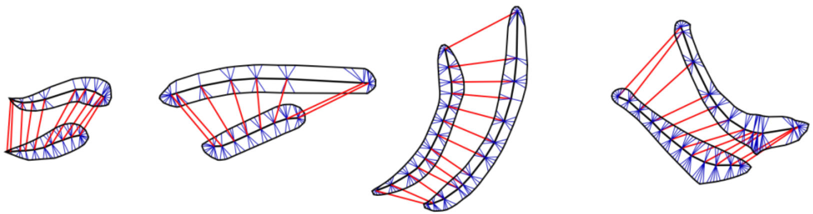


The automated inking system (cont’)

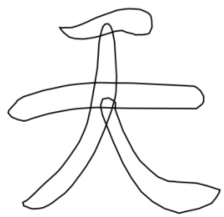
Online



B.1 Input B.2 Component analysis B.3 Library matching



B.4 & B.5 Correspondence and skeletal deformation



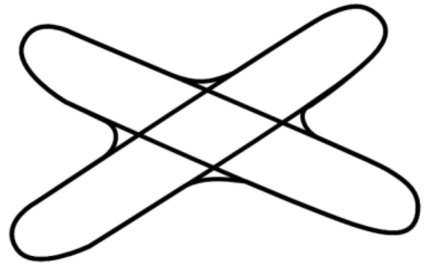
B.6 Output

Composition of strokes

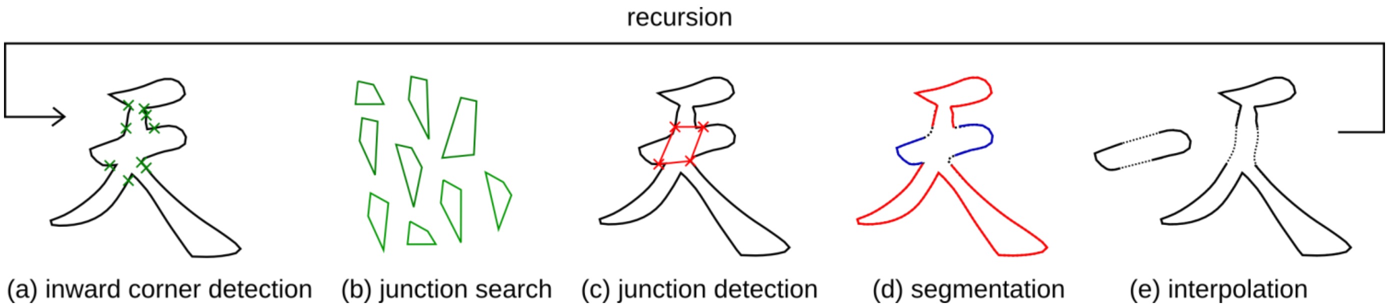
- composition is the union of two or more intersecting strokes

Assumptions:

- no overlap or end to end connection
- composition must yield identifiable junctions (but junction artifacts are allowed)
- associative: a composition of several composites is also a valid composite
- commutative: order of adding strokes does not matter

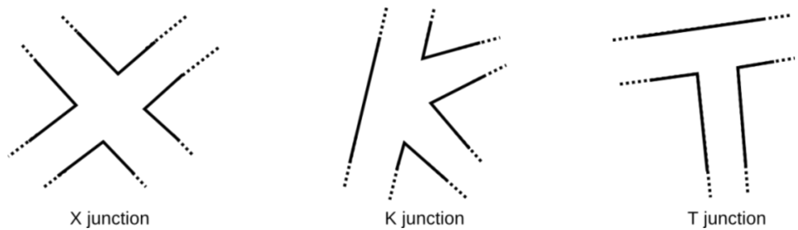


The decomposition algorithm



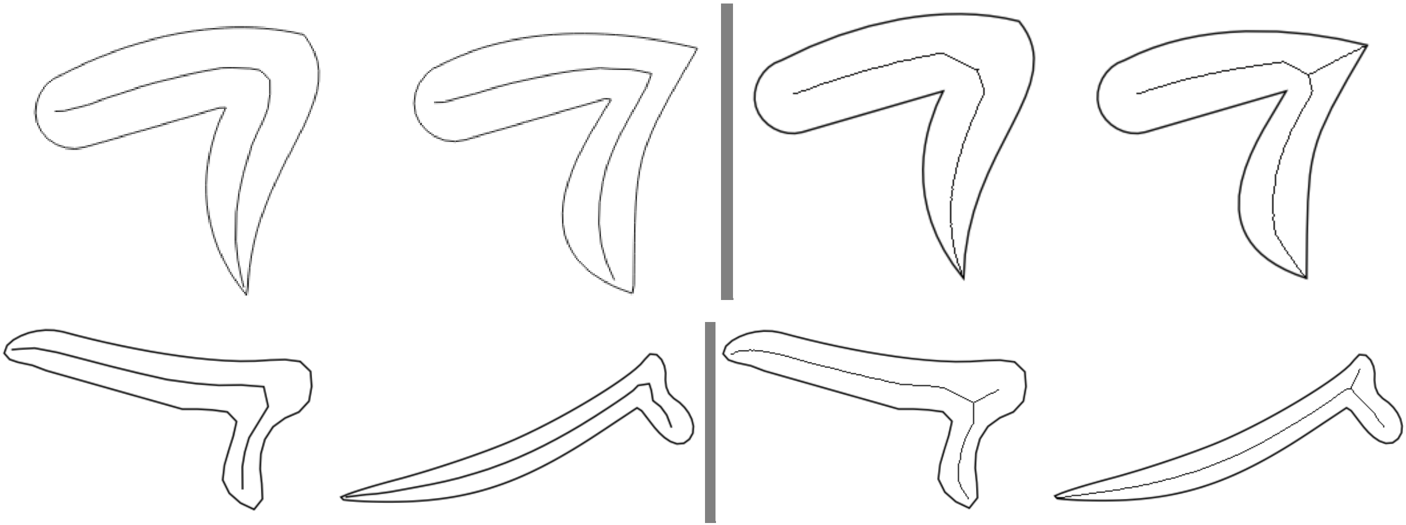
Visual summary of the decomposition algorithm

- recursive on the grand scale
- contour is analyzed for inward corners
- several corners are selected to form junctions



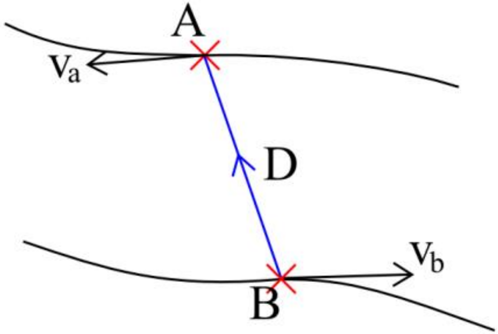
Thinning

Contour thinning vs morphological thinning

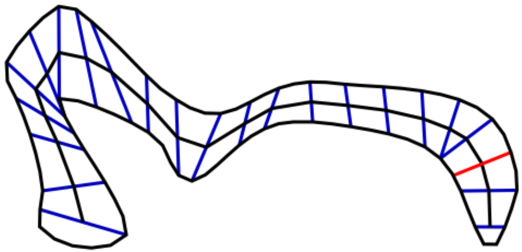


Left: our method; Right: morphological thinning

The thinning algorithm



Pairing up points on the contour



Thinning of a stroke. Pairs rendered in blue.

Skeletal stroke representation and deformation

- Do not confuse the name “skeletal stroke representation” with “skeletal strokes”
- A stroke transfer model without intermediate representation

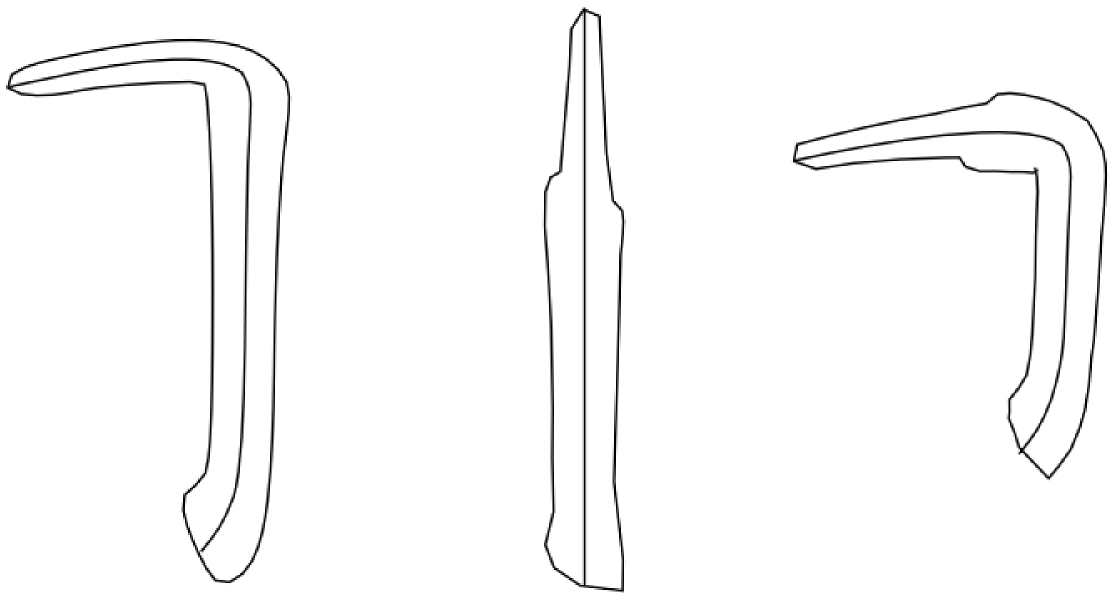
✓ Preserves the stroke characteristics

✓ Supports a larger set of (expressive) strokes

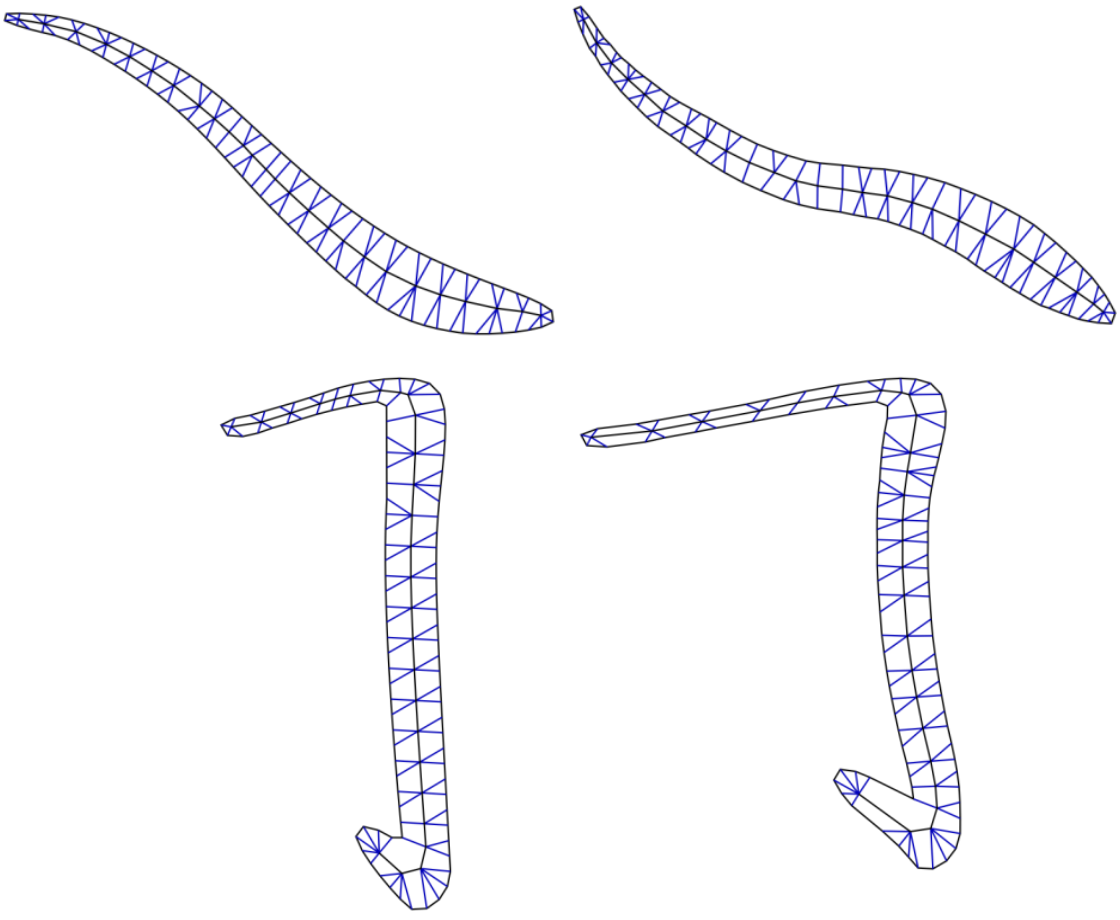
✘ Limited range of deformation

See P.9

Transfer of skeletal strokes



Left: original stroke, Center: straightened figure, Right: deformed stroke

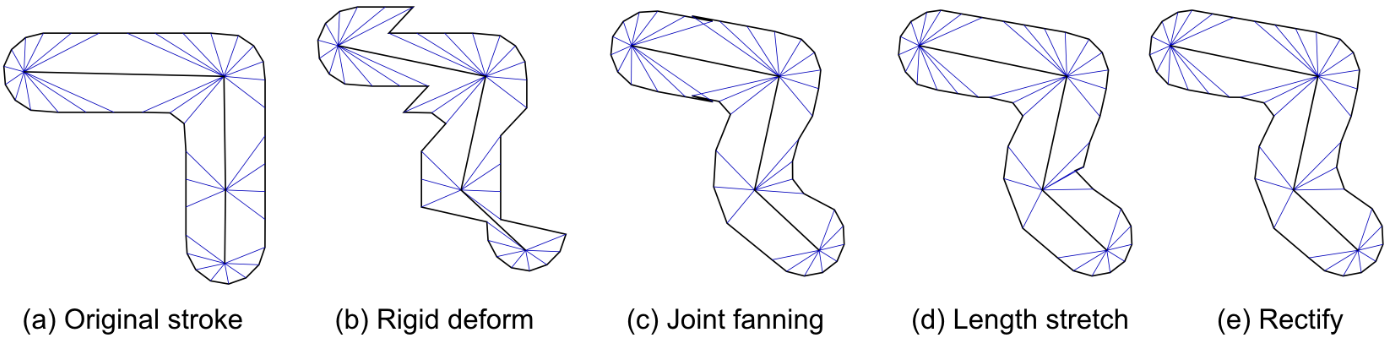


Left: Original skeletal representation. Right: deformed stroke

Skeletal stroke deformation

✓ Single iteration computation, extremely efficient

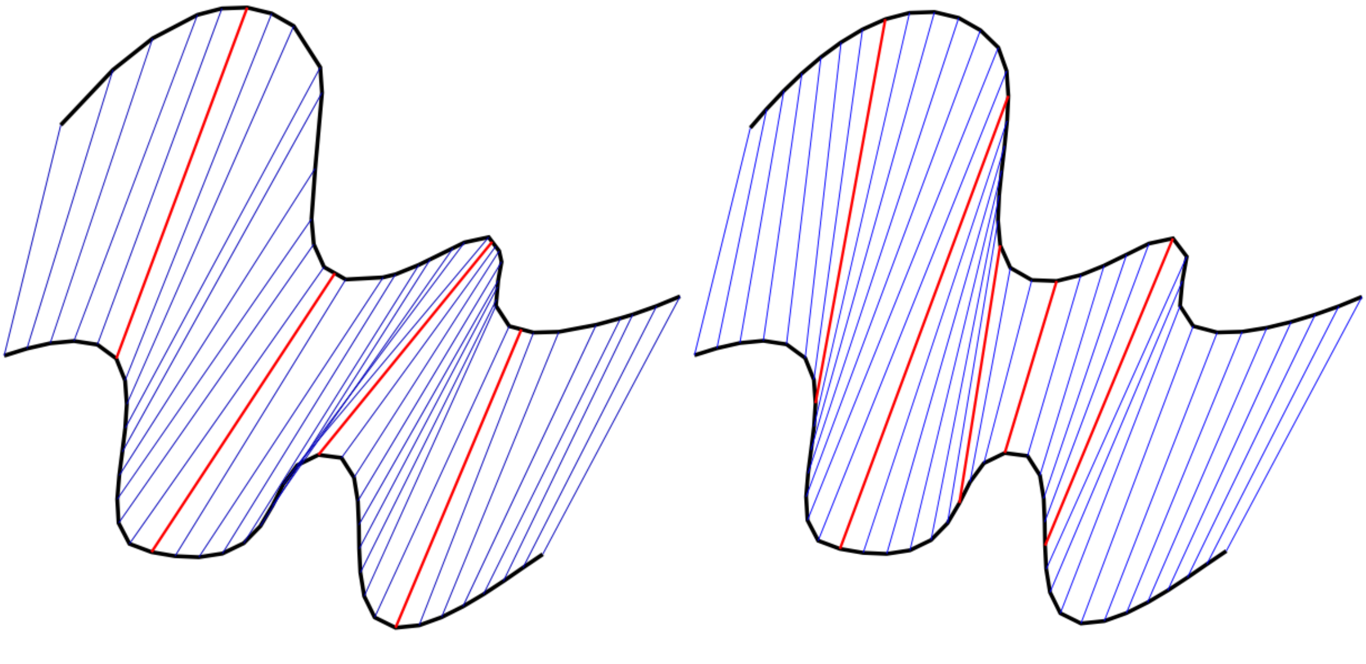
✘ Less “elastic” then energy minimization based algorithms



A visual summary of the deformation algorithm

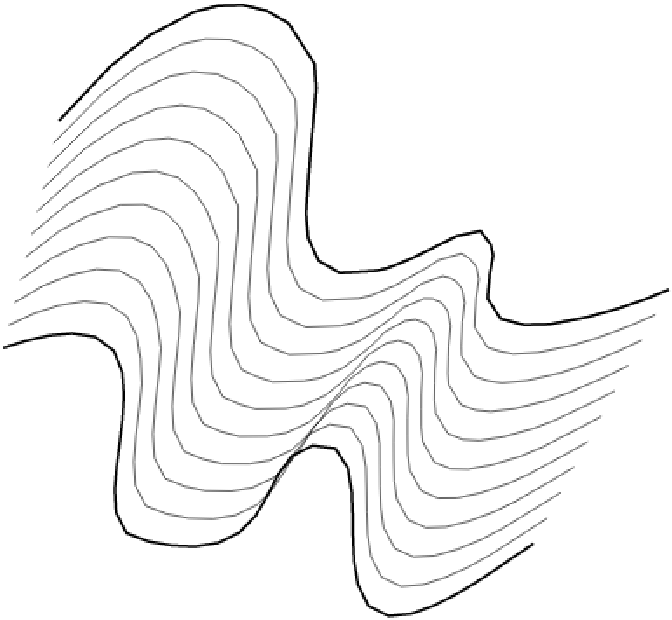
Curve correspondence

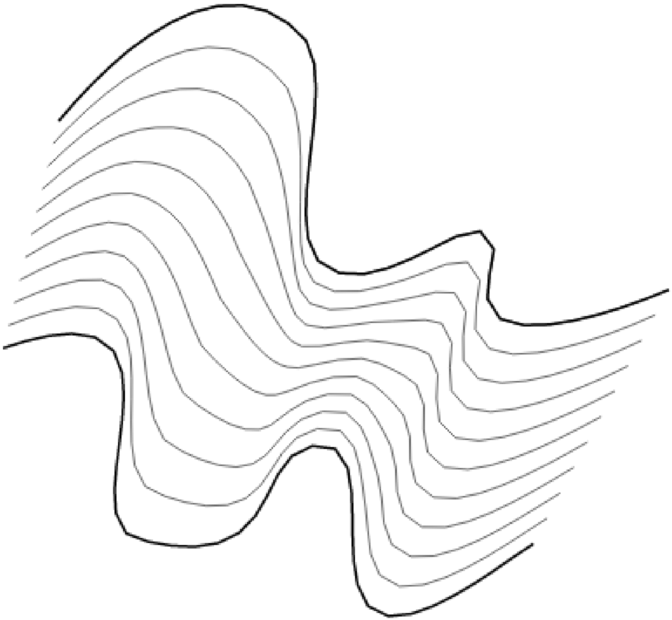
- Naive method simply corresponds according to path length
- Which ignores the characteristics of the curves



Left: our result. Right: naive correspondence

Curve interploation

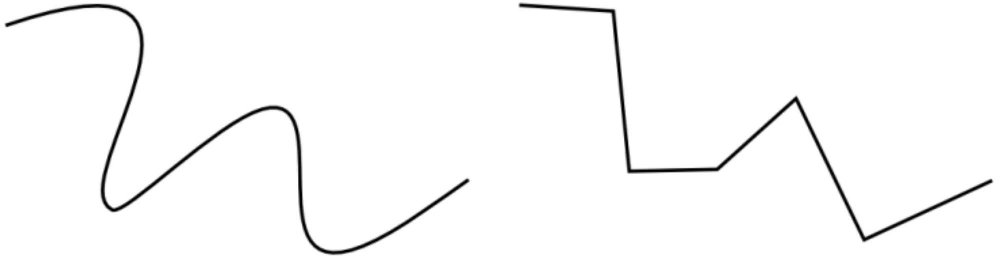




Left: our result. Right: naive

The curve correspondence algorithm

1. Curves are simplified to a characteristic polyline



2. Correspondence. For each point on one polyline, the cost to every points of the other polyline is computed and the point with the lowest cost is chosen:

$$5 \cdot | \, l[i] - l[j] \, | + | \, \theta[i] - \theta[j] \, | + | \, \nabla\theta[i] - \nabla\theta[j] \, |$$

where i, j are indices of polyline A and B respectively. $l[x]$ denotes the normalized path length at x. θ denotes the angle of the velocity vector. $\nabla\theta$ denotes the discrete curvature.

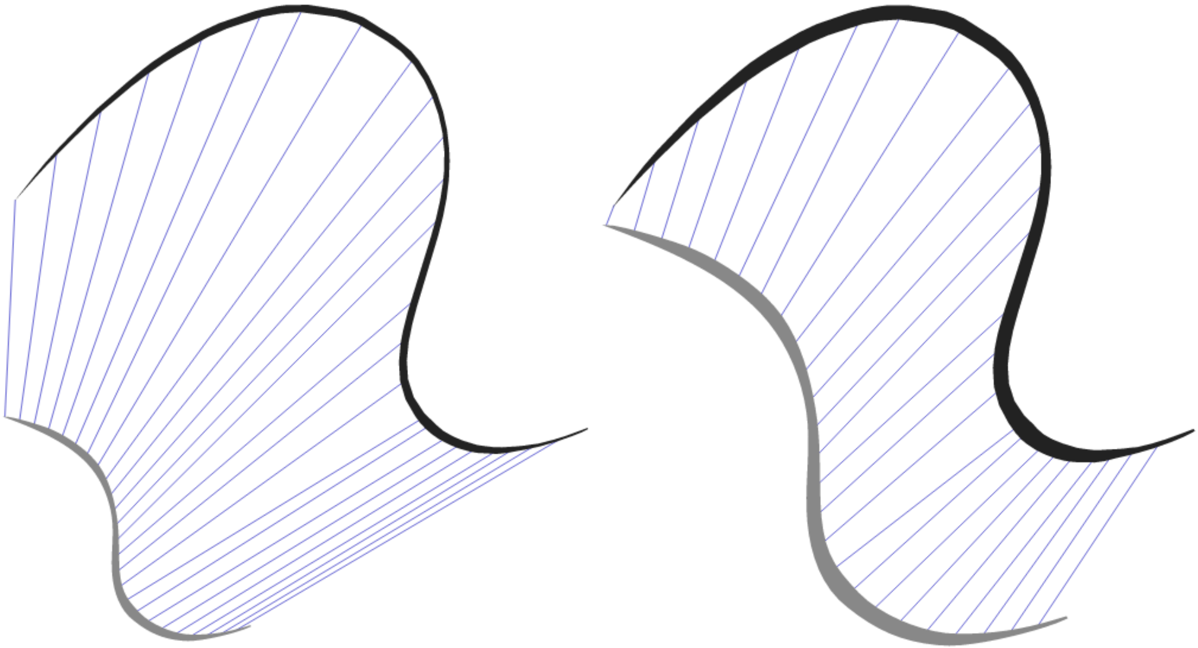
3. Resampling

Stroke transfer

Given a sample stroke and an input curve, the goal is to transfer the characteristics of the stroke to the curve. Stroke transfer has 3 steps:

1. sample stroke is thinned and the skeletal representation is computed
2. correspondence between the central axis and the input curve is computed
3. skeletal stroke deformation

Thickness control



Result of stroke transfer (black stroke). On the right, the sample stroke (gray) is scaled up before deformation. Correspondence is rendered in blue.

Stroke interpolation

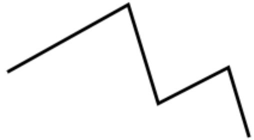
Stroke intersections and interactions

An inked illustration. Right: some group patterns

Some components in Chinese calligraphy

Component profile

Path



The characteristic polyline as explained in P.27

Hooks

A hook means a short, sharp turning segment at the end of a polyline

Dot

A dot is a very short line segment

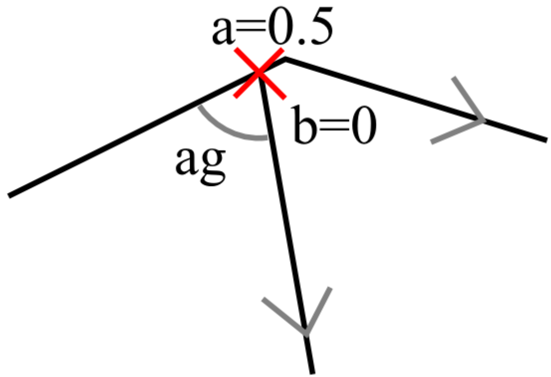
Bound

Bounding box { xmin, xmax, ymin, ymax}

Intersects

The intersections of a polyline with all others in the same group are computed. It is an array of the following data structure:

- **a,b** is the normalized path length of where the intersection occurs on characteristic polyline A and B respectively
- **ag** is the included angle at the intersection
- **i** is the id of B in the group



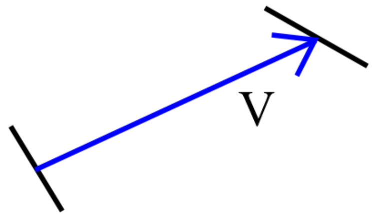
Joins

A join means that a curve's end is very close to the end of another curve.

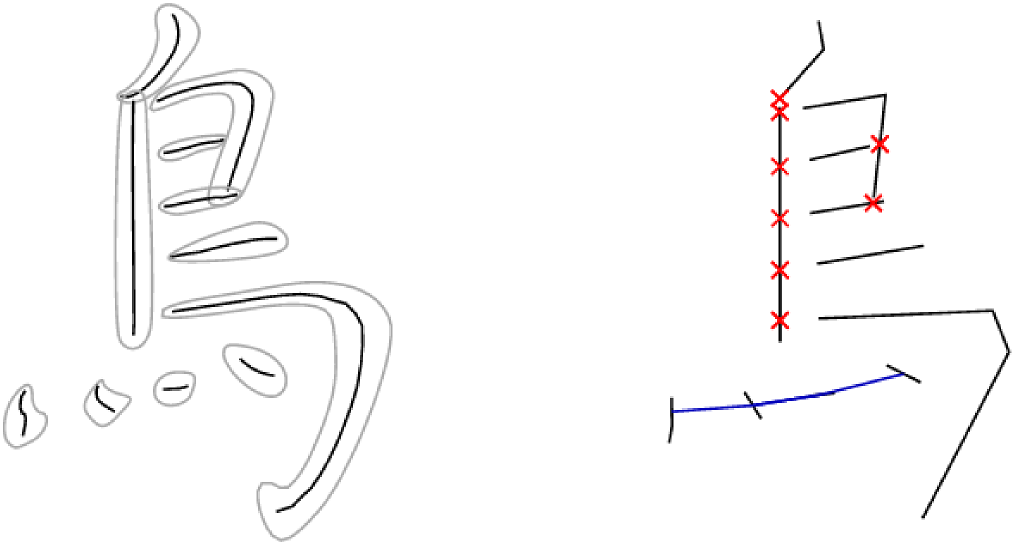
Other dots

Dots are usually isolated. They often have spatial interactions with other dots. It is an array of the following data structure:

- **V** is the relative position of the other dot
- **i** is the id of the other dot in the group



Example



Component profile. Stroke and path (left), characteristic polyline (right). Intersections are marked as red crosses and dot proximities in blue lines.

Component similarity

Component similarity is a numerical measurement of the difference between two component profiles. It constitutes of the following:

Correspondence cost

Of the two component path (characteristic polyline)

Bound error

The bound error is composed of 2 terms, aspect ratio and area error:

$$| A.width/A.height - B.width/B.height | +$$

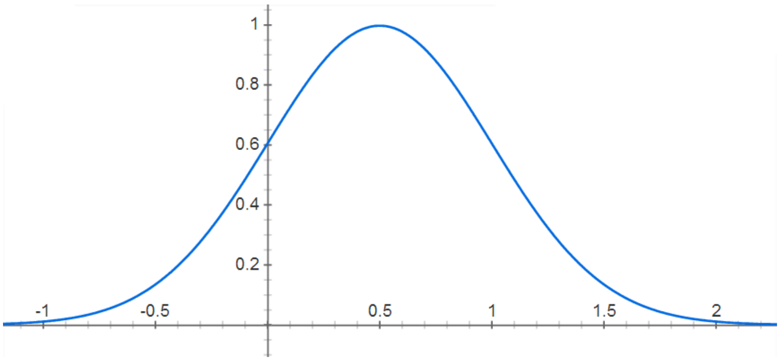
$$| A.width \cdot A.height - B.width \cdot B.height |$$

Intersects

Elements from the two sets are optimally matched and the assignment cost is taken, using the following cost function:

$$\text{Cost}(A,B) = 5\cdot|\ A.a\text{-}B.a\ |\cdot w(A.a) + |\ A.b\text{-}B.b\ |\cdot w(A.b) + |\ A.ag\text{-}B.ag\ |$$

$$\text{Where } w(x) = 1.25 \cdot \left(\frac{1}{\frac{1}{2}\sqrt{2\pi}}\right) \cdot \exp\left(-\frac{(x-0.5)^2}{2(0.5)^2}\right)$$



Graph of y=w(x)

Other dots

Elements of “other dots” are matched by the following cost function:

$$\mathrm{Cost}(A,B) = 10\cdot|\angle A-\angle B| + \big||A|-|B|\big|$$

The assignment cost is taken. Unmatched points add extra penalty.

Other differences

Others differences like dot, joins and hooks add certain penalty.

Weighted sum

The above values are weight-summed to a real number. The exact weightings vary according to the style of illustration and unit of measurement.

Optimal assignment

The Hungarian algorithm is used.

A=[1,4,9]

B=[3,5]

Optimal assignment = 1<->3, 4<->5

Assignment cost = 3-1+5-4 = 3

A=[1,4,9]

B=[3,7]

Optimal assignment = 3<->4, 7<->9

Assignment cost = 4-3+9-7 = 3

Component matching

- Given a query component profile, it is searched linearly in the library for the 5 candidates with the highest similarity
- To attain a maximal sub-graph match heuristically, candidates which are neighbors in both the query graph and library graph space are promoted
- After that, the best match is selected for stroke transfer

Evaluation

Methodology

- Test on real world data
- Illustrations drawn by artists were collected

  (All artwork in this presentation are reproduced without permission)
- Subjective survey

Two styles of illustration

- Chinese calligraphy
- Cartoons

Artwork used

Chinese calligraphy

Artwork courtesy萬承毅

人皇始制文字乃服衣裳推位讓
國有虞陶唐弔民伐罪周發殷湯
坐朝問道垂拱平章愛育黎首臣
伏戎羌遐邇一體率賓歸王鳴鳳
在樹白駒食場化被草木賴及萬
方蓋此身髮四大五常恭惟鞠養
豈敢毀傷女慕貞絜男效才良知
過必改得能莫忘罔談彼短靡恃

Written by unknown

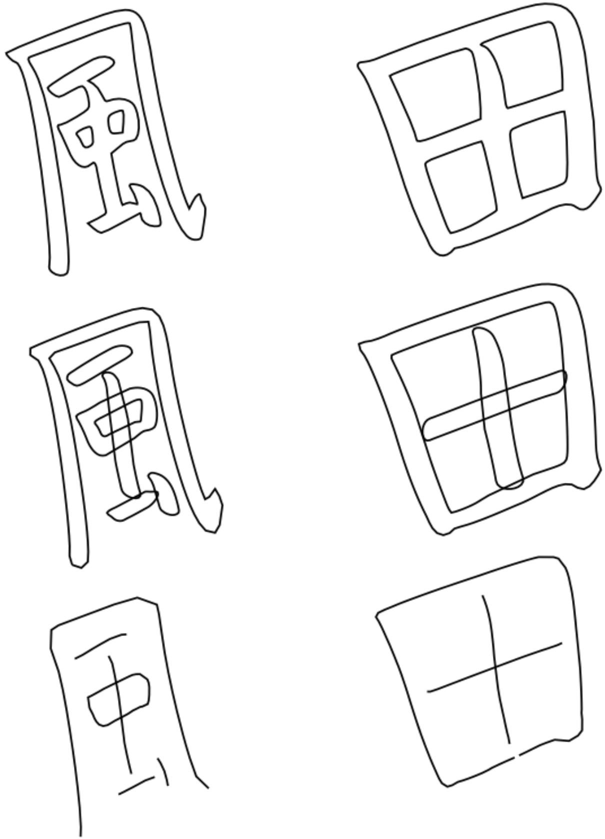
Cartoons

Artwork courtesy AnarchyWulf

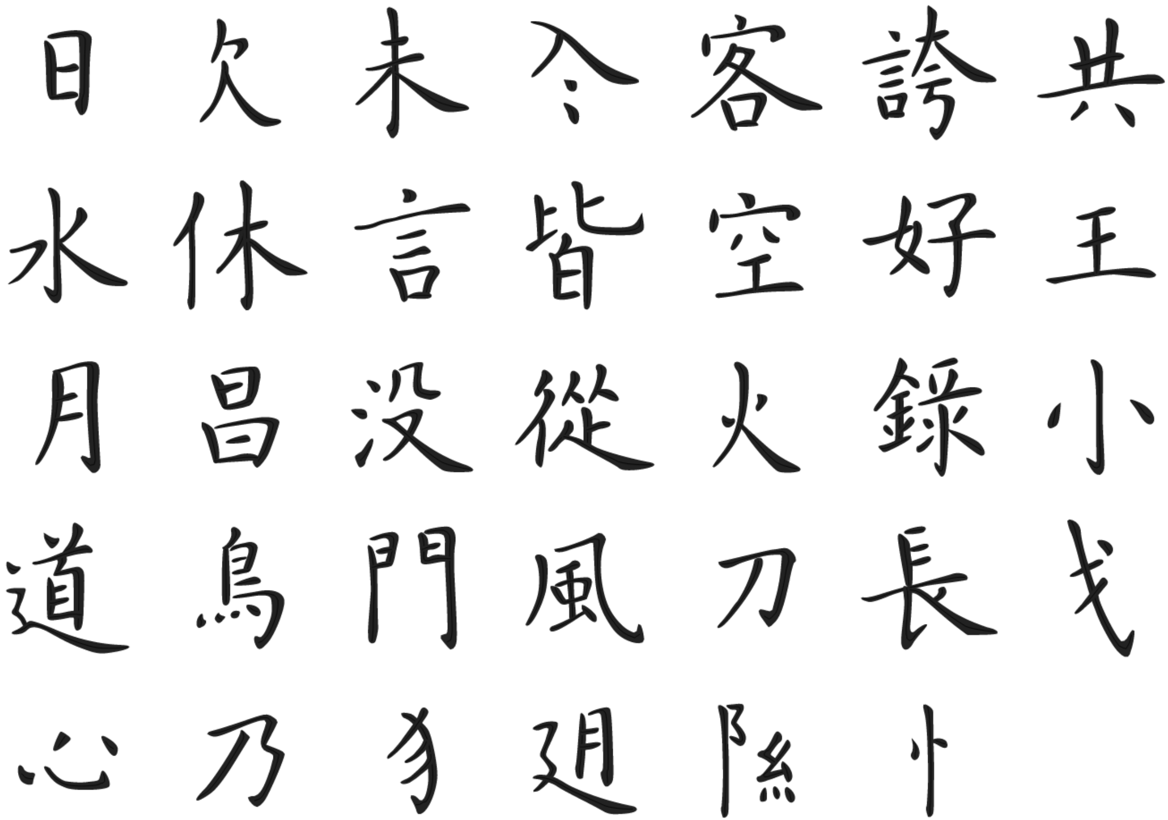
Frames extracted from the animation ***Toy Tinkers***, 1949 by Walt Disney Productions.

Decomposition and thinning

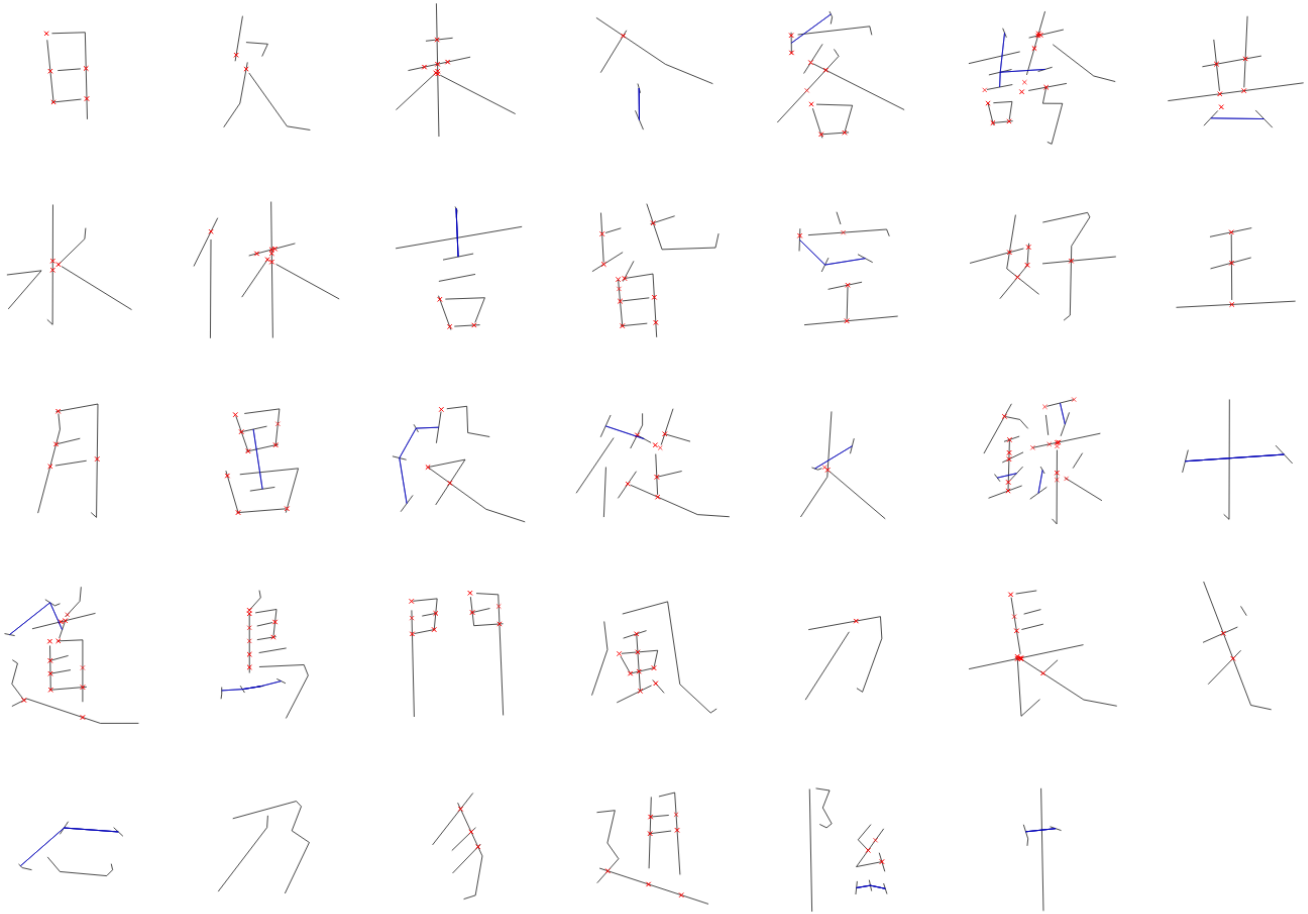
Hollow shapes



Stylization of Chinese calligraphy



Component library



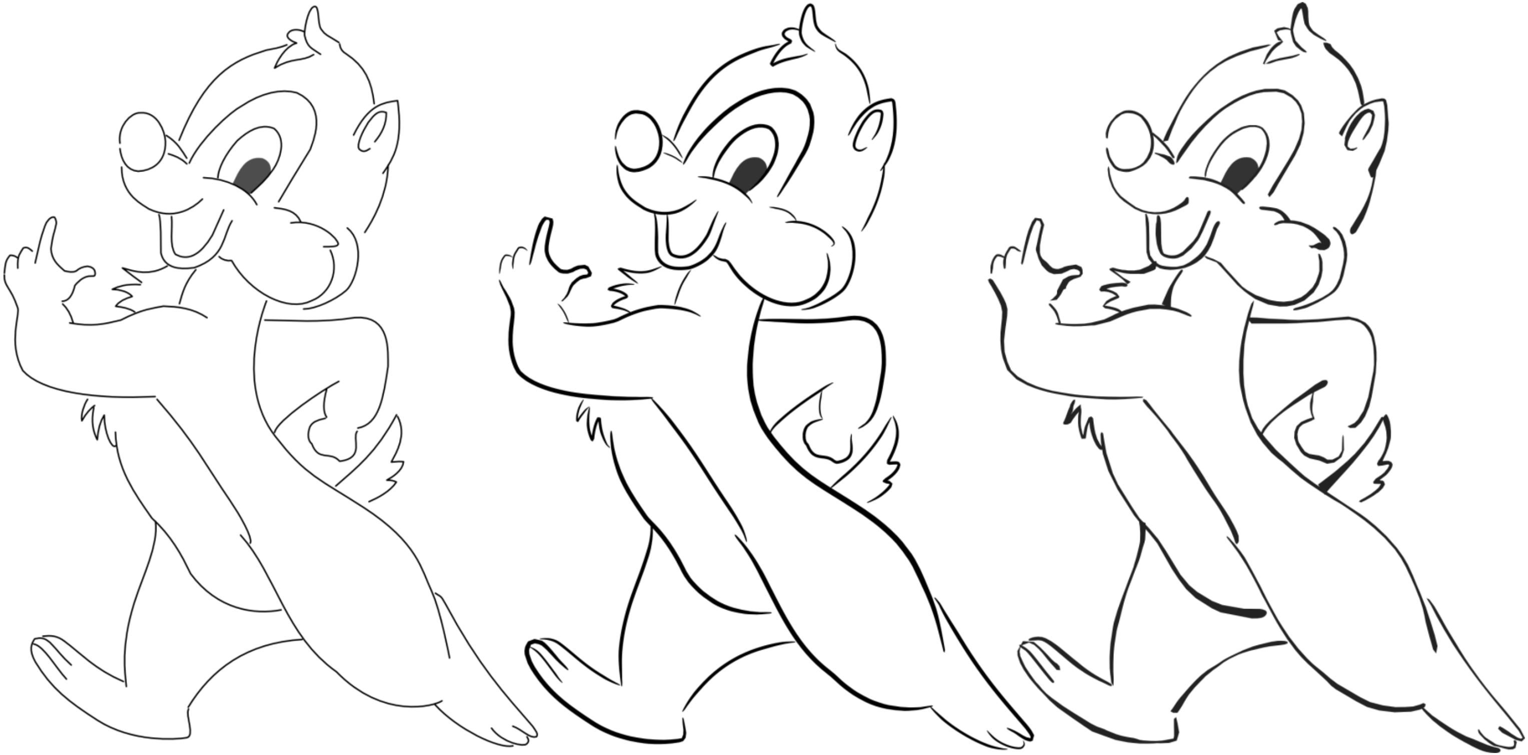
The corresponding component profiles

Input (top row), stylized result (middle), original characters (bottom)

Input curves and the stylization result

Stylization of cartoons

Cartoon stroke library



Input (left), basic stylization (middle), cartoon-stylized result (right). The solid eyes were added afterwards for aesthetic purpose.

Subjective survey

Which side is drawn by artist, which side is computer generated?

In this image, one row was computer generated, while the other was scanned from a calligraphy. Can you guess?

In this image, one column was computer generated, while the other was scanned. Can you guess?

- Participants were asked to state the quality of each artwork. The average rating was between **“good” and “very good”**.
- Participants were unable to judge which cartoon was computer generated (correct rate close to 50%).
- Participants were able to judge correctly which Chinese characters were generated, up to 90%.
- They were told the truth, and asked to rate the plausibility when compared to that drawn by artists. The average rating was **“plausible”**.

Quality	Extremely bad	Very bad	Bad	Quite bad	Average	Quite good	Good	Very good	Excellent
Plausibility	Unquestionably fake	Obviously fake	Fake	Questionable	Okay	Quite plausible	Plausible	Believable	Unbelievably true

Rating table

Artisticity

- The automated inking system is completely mechanical
- No element of machine learning or stochastic process
- Inking is mechanical, predictable and not fuzzy
- Inking is more a science than Art

Future applications

Font characters generation

- Far less Chinese fonts compared to alphabetic languages
- Chinese fonts have to support as much as 50,000 characters
- Regular fonts have been generated in the past
- Yet there was no successful example of calligraphic font generation known to us
- The automated inking system could be an integral part of a calligraphic font generation system

2D Animation

- Many animation tools had keyframe and tweening functionalities, but they were unable to satisfy artists' needs
- Every frame, stroke and detail was artists' hard work
- Most commercial production today utilized only fixed width strokes
- The ideas and techniques developed in the project could be further developed into a system that can truly automate the artistic workflow

Summary

- Addressed existing limitations
- Developed a stack of algorithms
- Stated the importance of stroke interactions
- Presented a novel data-driven model
- Demonstrated automated inking is achievable

Take away message

- Inking is a fundamental form of art
- Subtle details matter

Q & A

Acknowledgements

Photo taking

Shape theory in 2D

Classification	Interior	Exterior
Graphic pipeline	Raster	Vector
Concerns	Area	Contour
Algorithms	Morphological image processing	(we call them) Contour based shape processing

Rundown

0-7	Introduction (7 minutes)
7-22	Algorithms (15 mintutes)
22-32	Demonstration (10 minutes)
32-35	Conclusion (3 minutes)
35-40	Q&A (5 minutes)

Total: 40 minutes

<u>Afterwards</u>

Acknowledgement

Photo taking