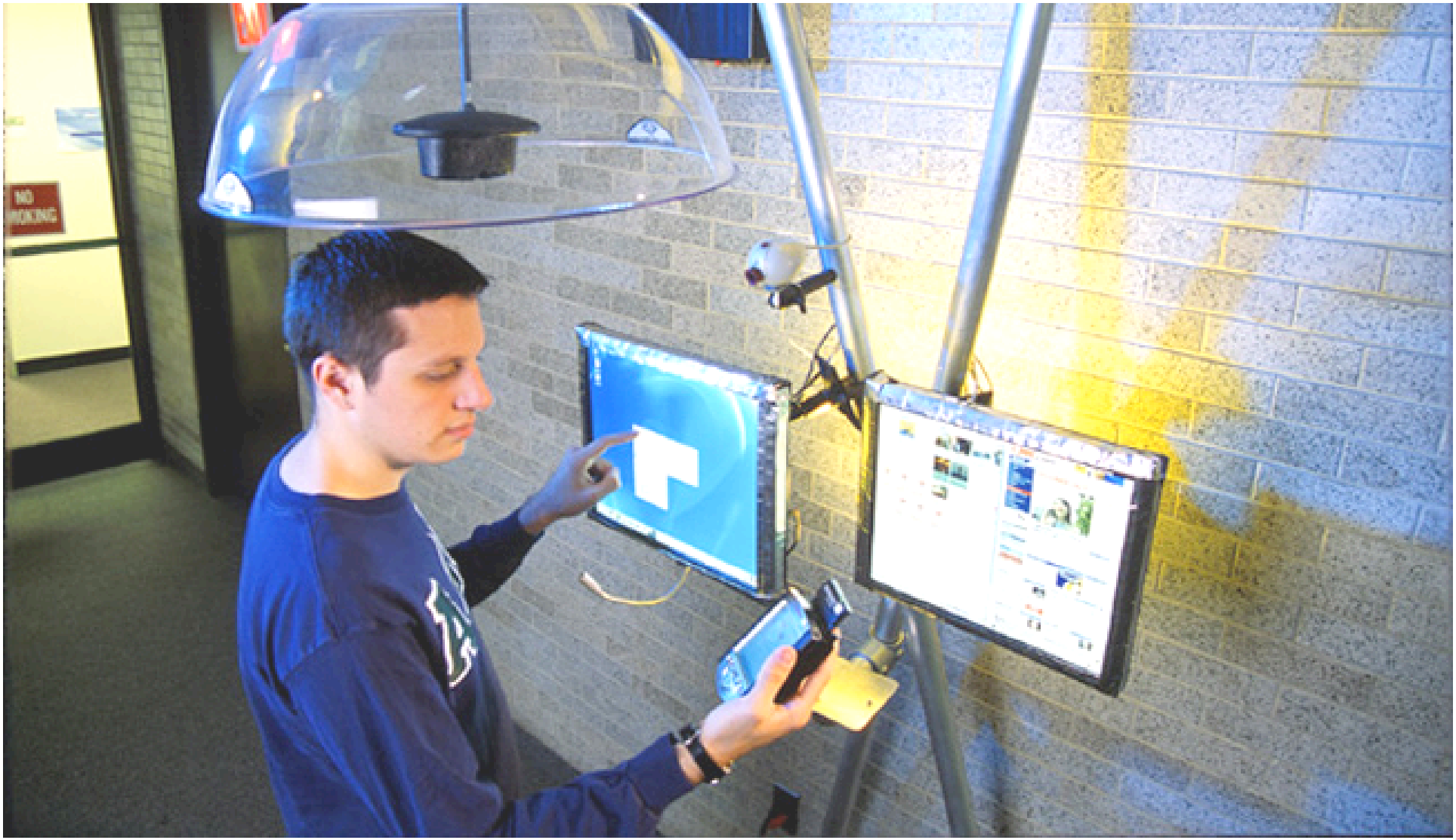
Distinctive-touch

gesture-based authentication for touch-screen displays

max van kleek : mas.622j : dec.7.2004



ok-net

"smart" digital touch-screen displays in places people congregate, accidentally meet one another most often

located in high-traffic spaces: building entrances, elevator lobbies, hallways, lounges, kitchenettes, cafeterias,

personalization requires user identification

biometrics violate privacy - users can't change their biologies

RFID requires user to carry around a badge

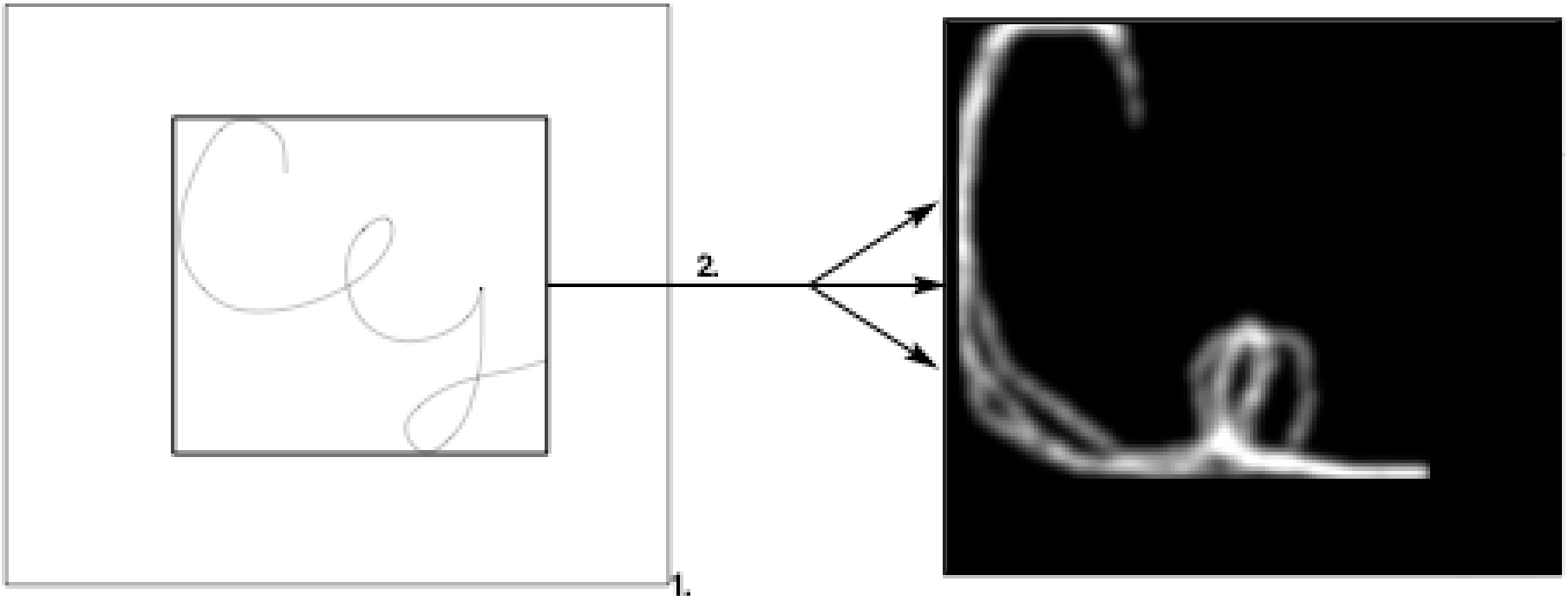
usernames/passwords are cumbersome, slow

these techniques are **overkill**, and not scalable

distinctive touch:

recognizes you by your “passdoodle”
a **multi-stroke** gesture-based signature

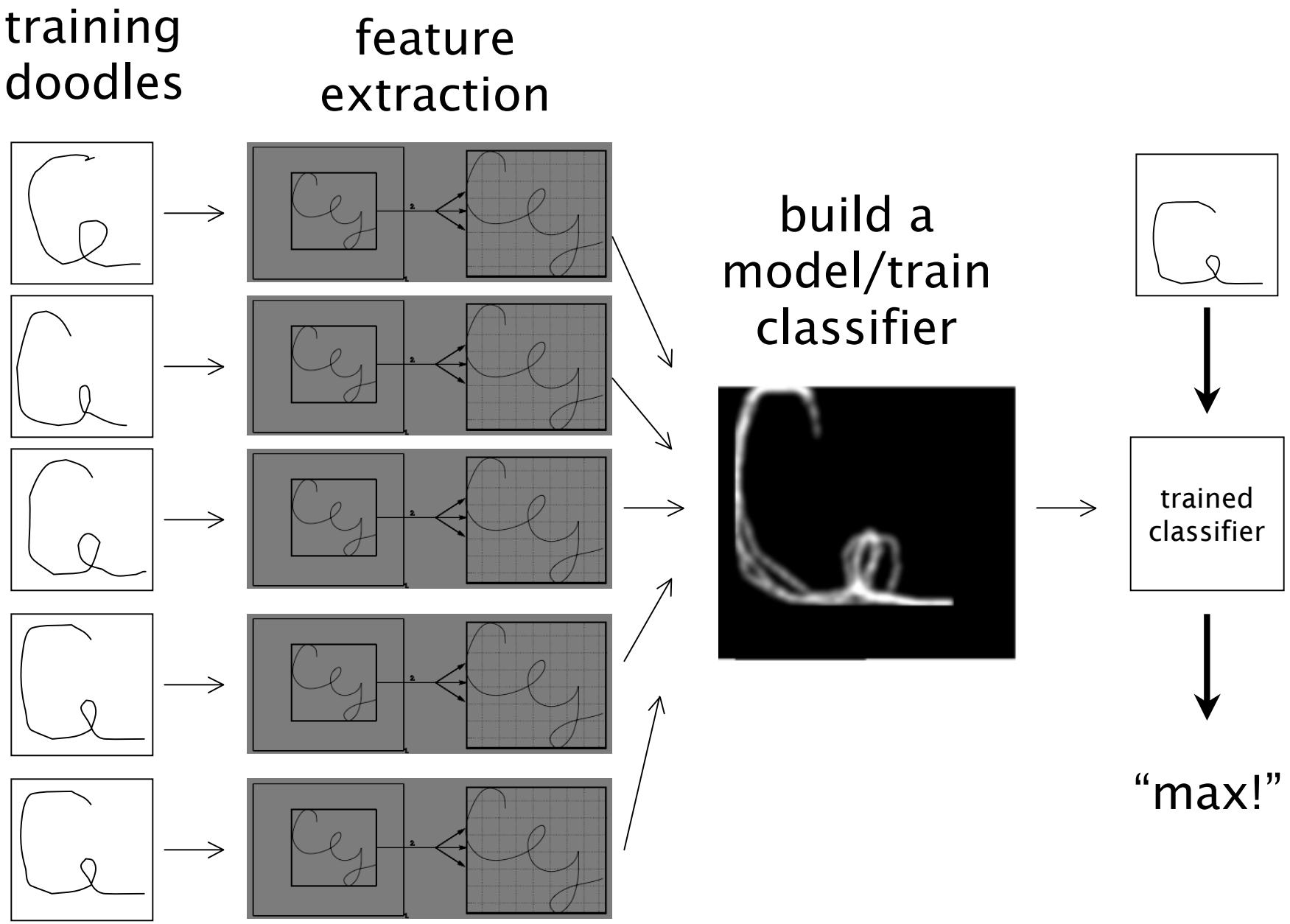
users train 5-10 training examples: < 5 seconds each



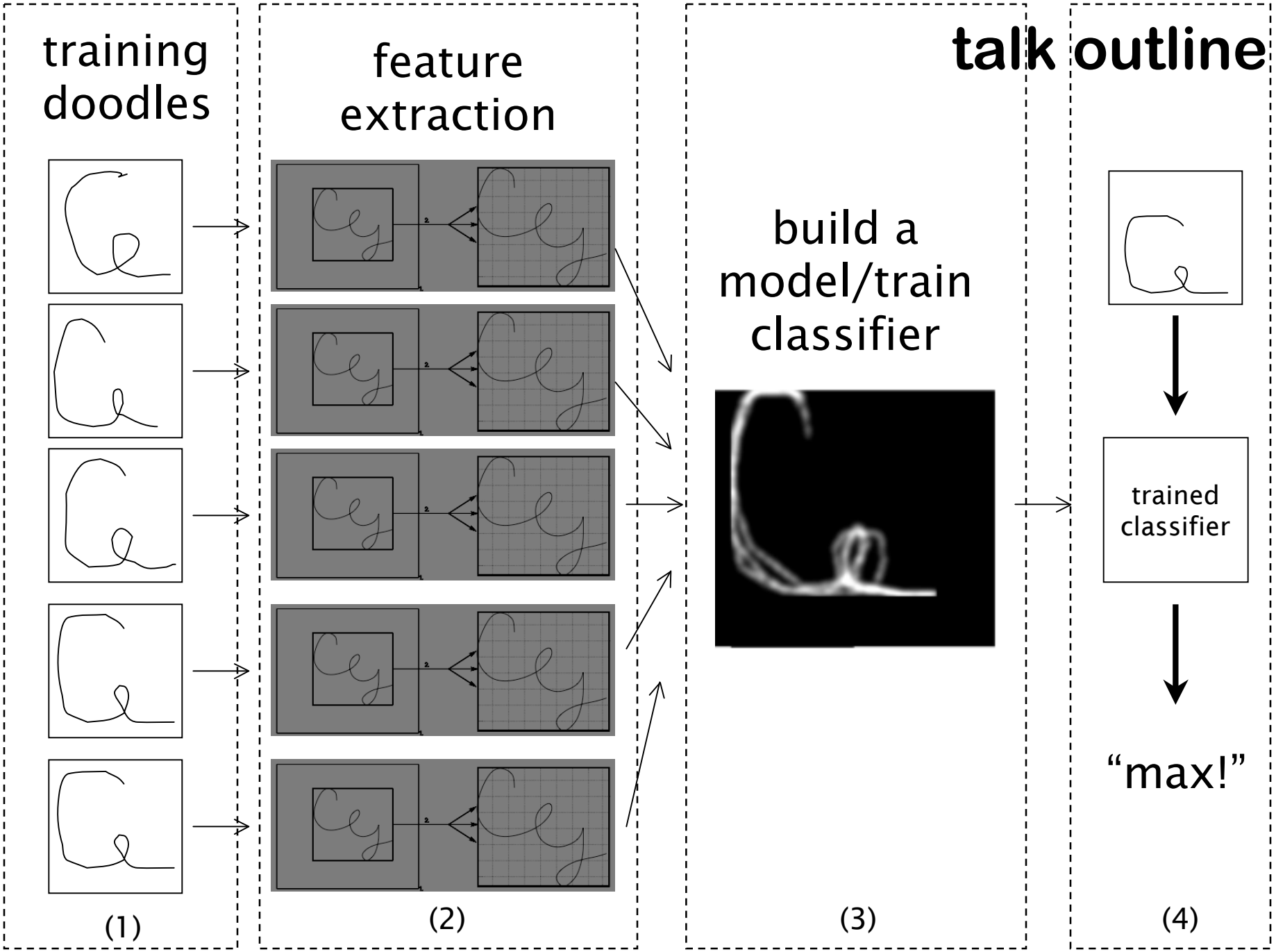
distinctive touch interaction times

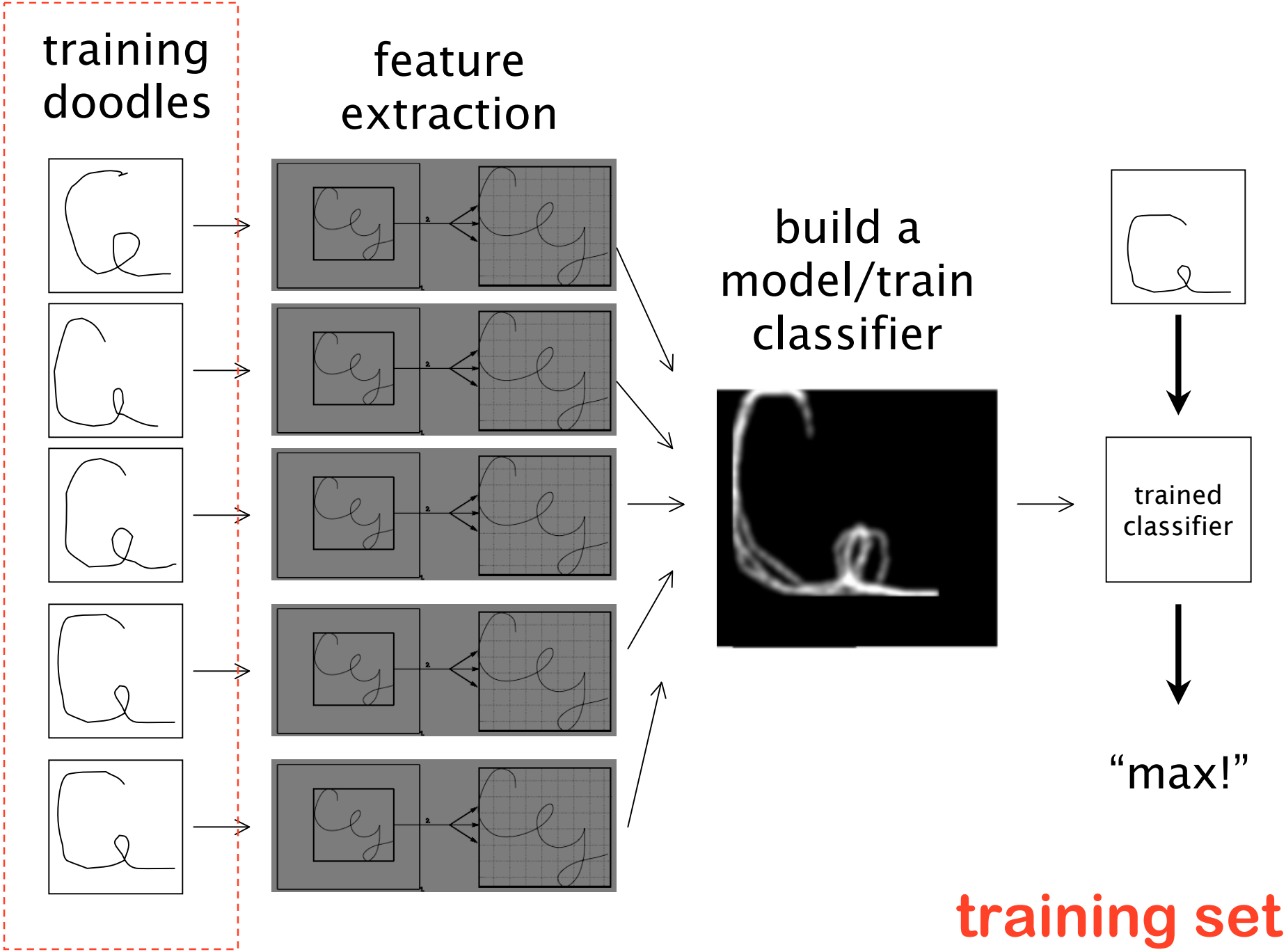
```
>> min(times)

1.0290    0.4060    1.5410    0.5180    0.3540    1.1390    1.3990
1.0310    2.2330    1.0930

>> mean(times)

1.3309    0.9380    2.4711    0.9367    0.5284    1.4328    1.5674
1.1913    2.4735    1.3355

>> max(times)

2.0140    1.5030    2.9120    1.3070    0.6900    1.8130    1.9180
1.3330    2.8690    1.7490

>> std(times)

1.2539    0.3313    1.1927    4.3841    2.1382    0.3453    2.2351
0.3209    0.7660    1.6609

>> mean(mean(times))

1.4206

>> vstd(times)

0.6467
```

< 5 seconds

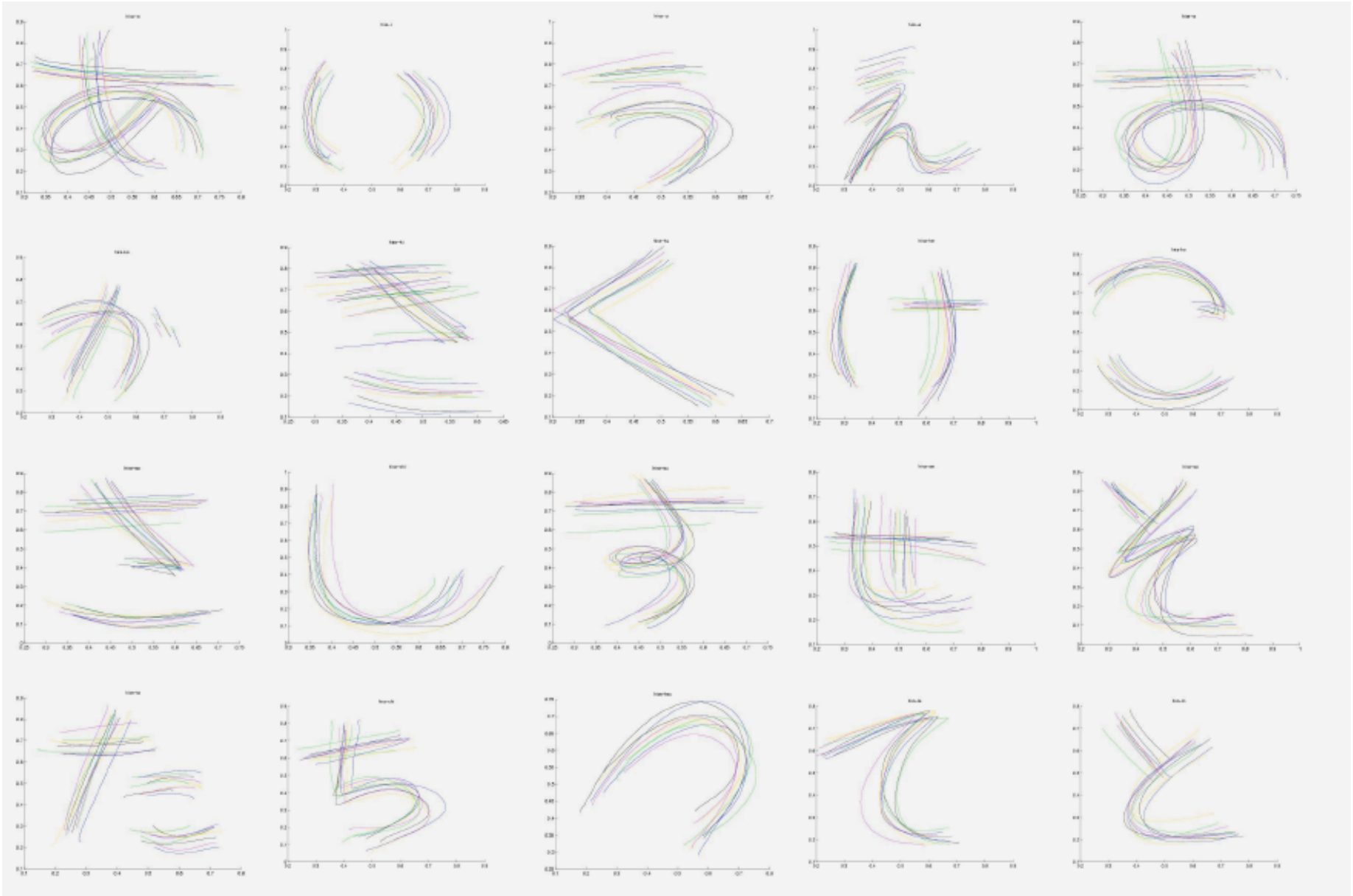


building a dt classifier

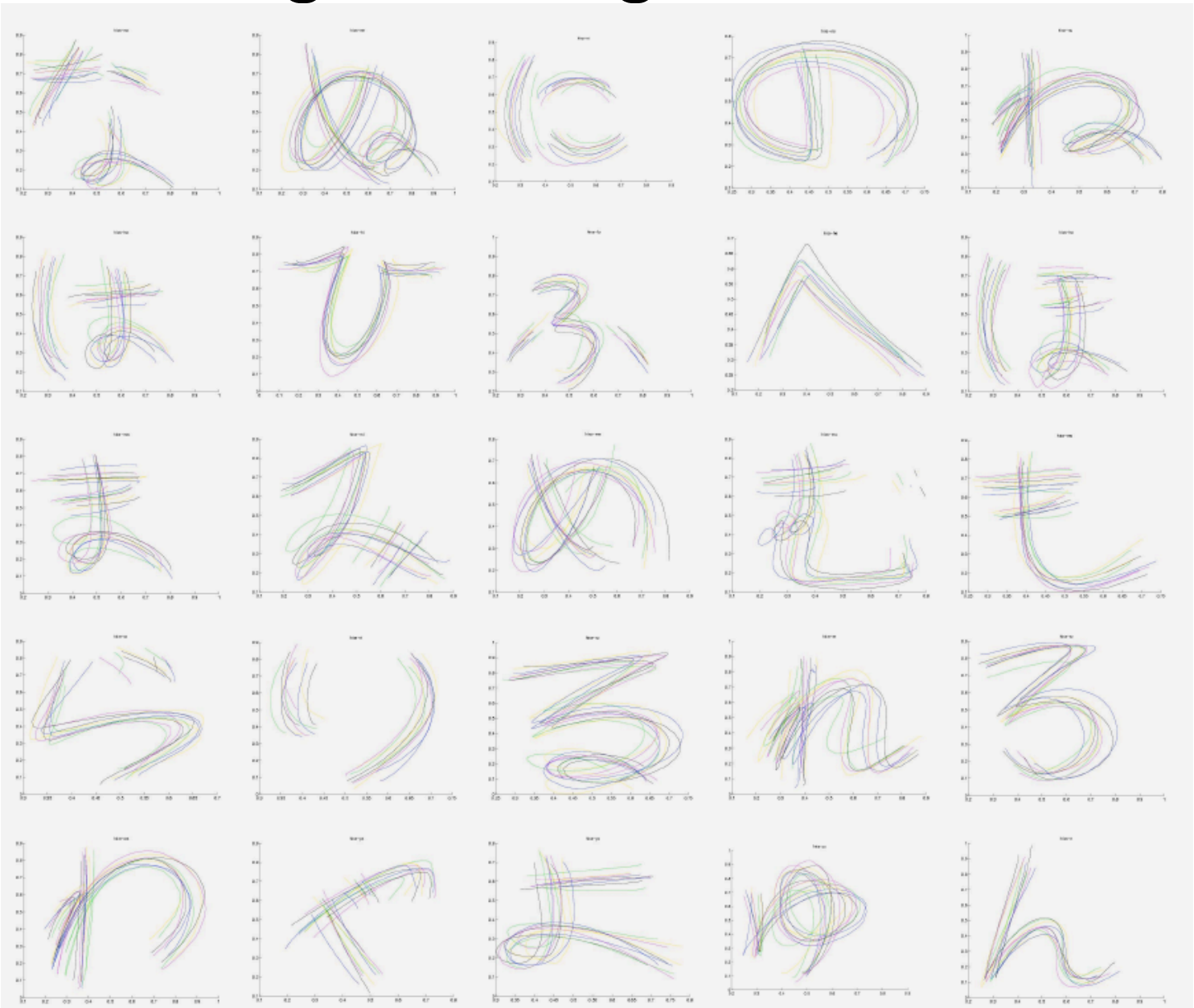

training doodles
feature extraction
build a model/train classifier
talk outline
trained classifier
"max!"
(1)
(2)
(3)
(4)


training doodles
feature extraction
build a model/train classifier
trained classifier
"max!"
training set

training set: hiragana character set



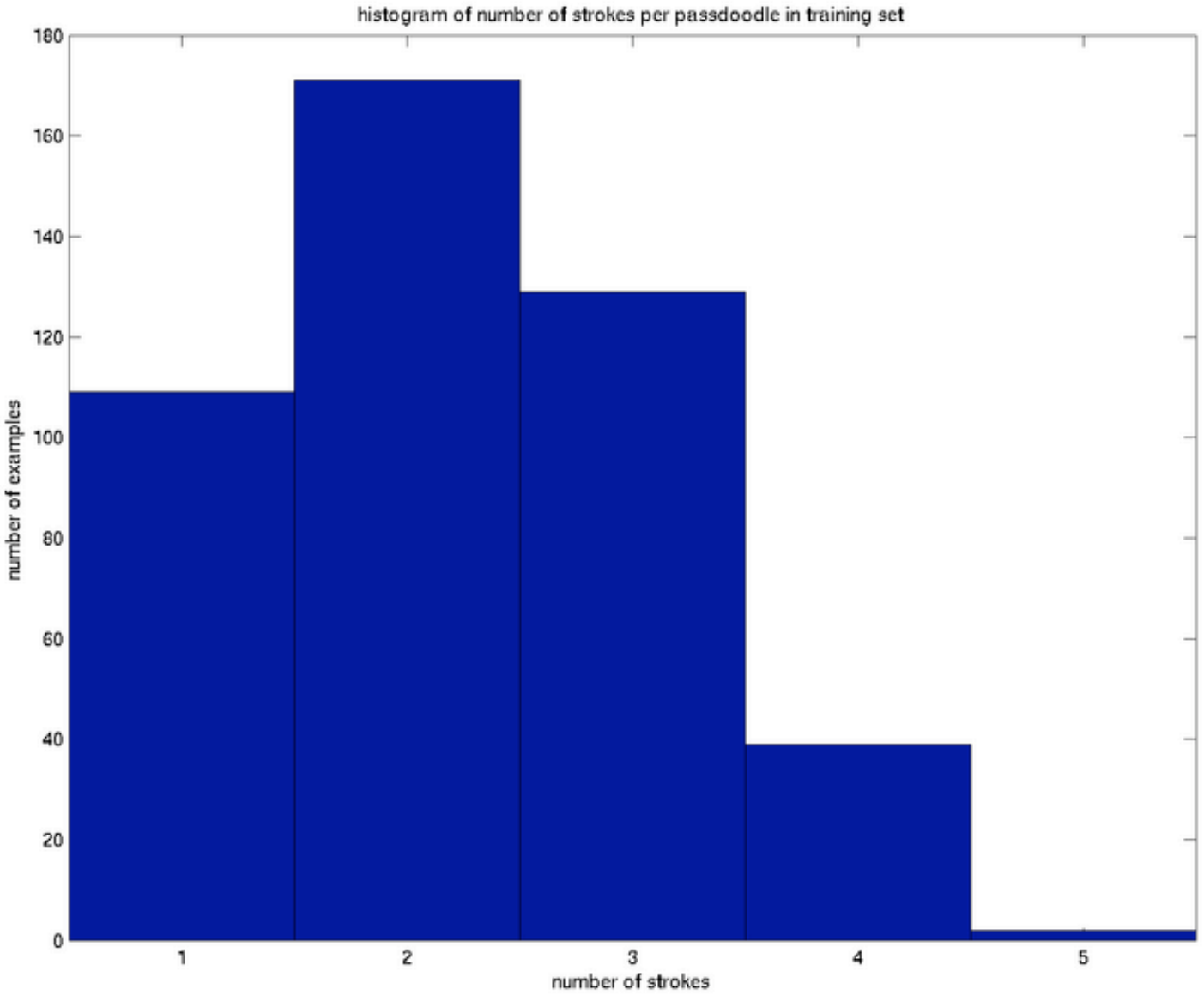
training set: hiragana character set



train set:
45 classes
10 ex each
1-5 strokes

test set:
45 classes
5 ex each

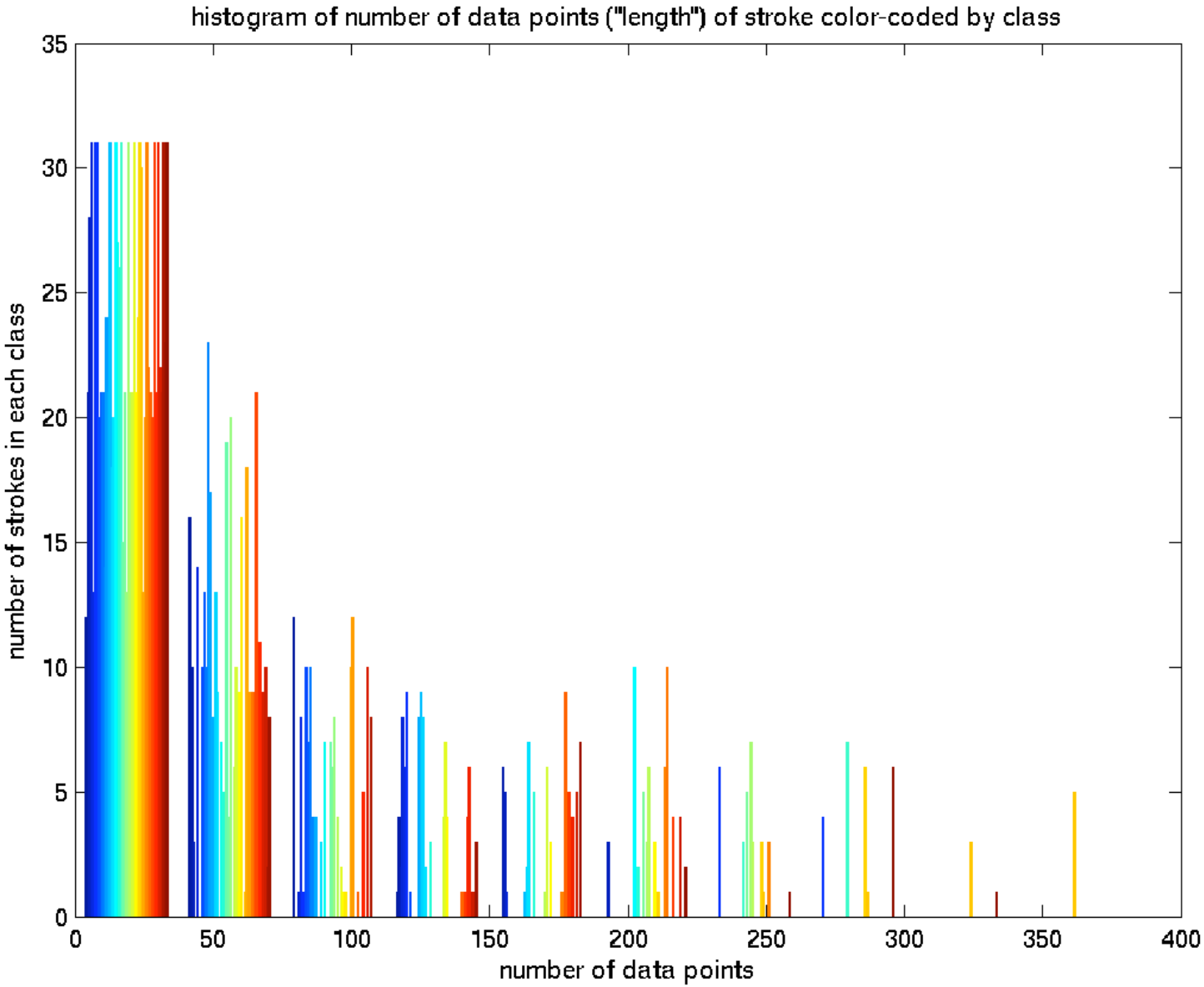
training set: hiragana character set



train set:
45 classes
10 ex each
1-5 strokes

test set:
45 classes
5 ex each

training set: hiragana character set

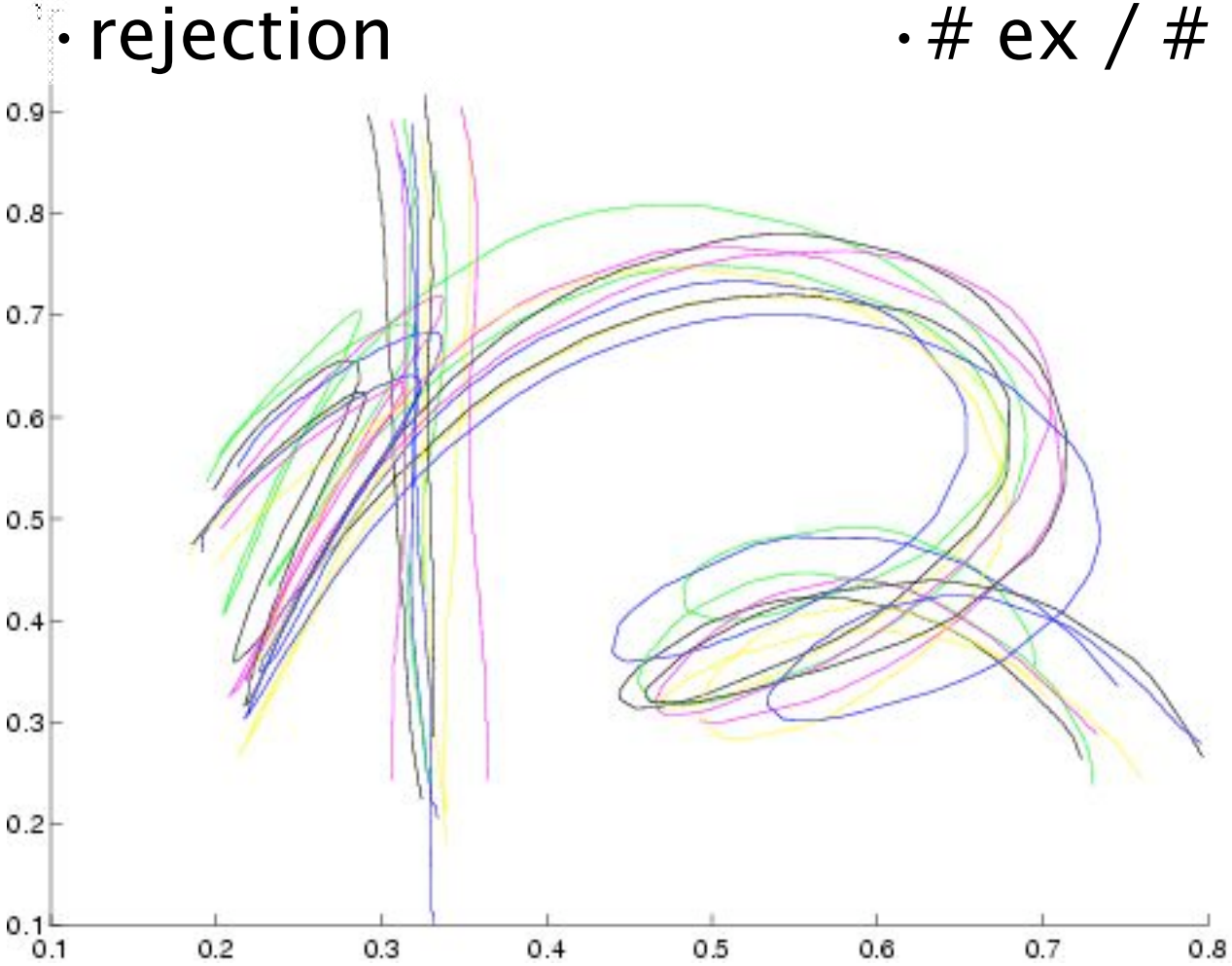


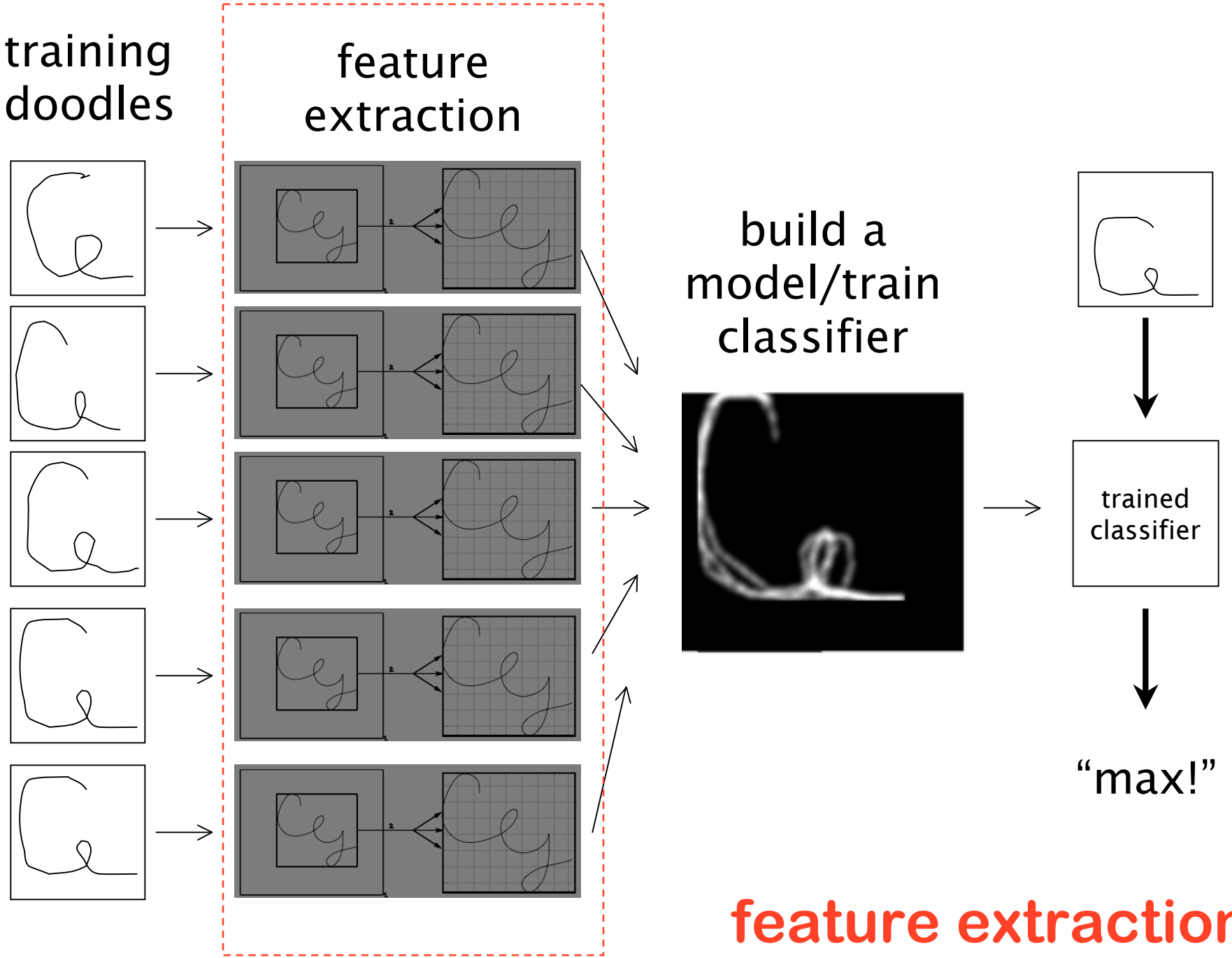
train set:
45 classes
10 ex each
1-5 strokes

test set:
45 classes
5 ex each

dt versus handwriting recognition

- stroke order
- rejection
- velocity and timing
- # ex / # of classes




training doodles
feature extraction
build a model/train classifier
trained classifier
"max!"
feature extraction

feature extraction

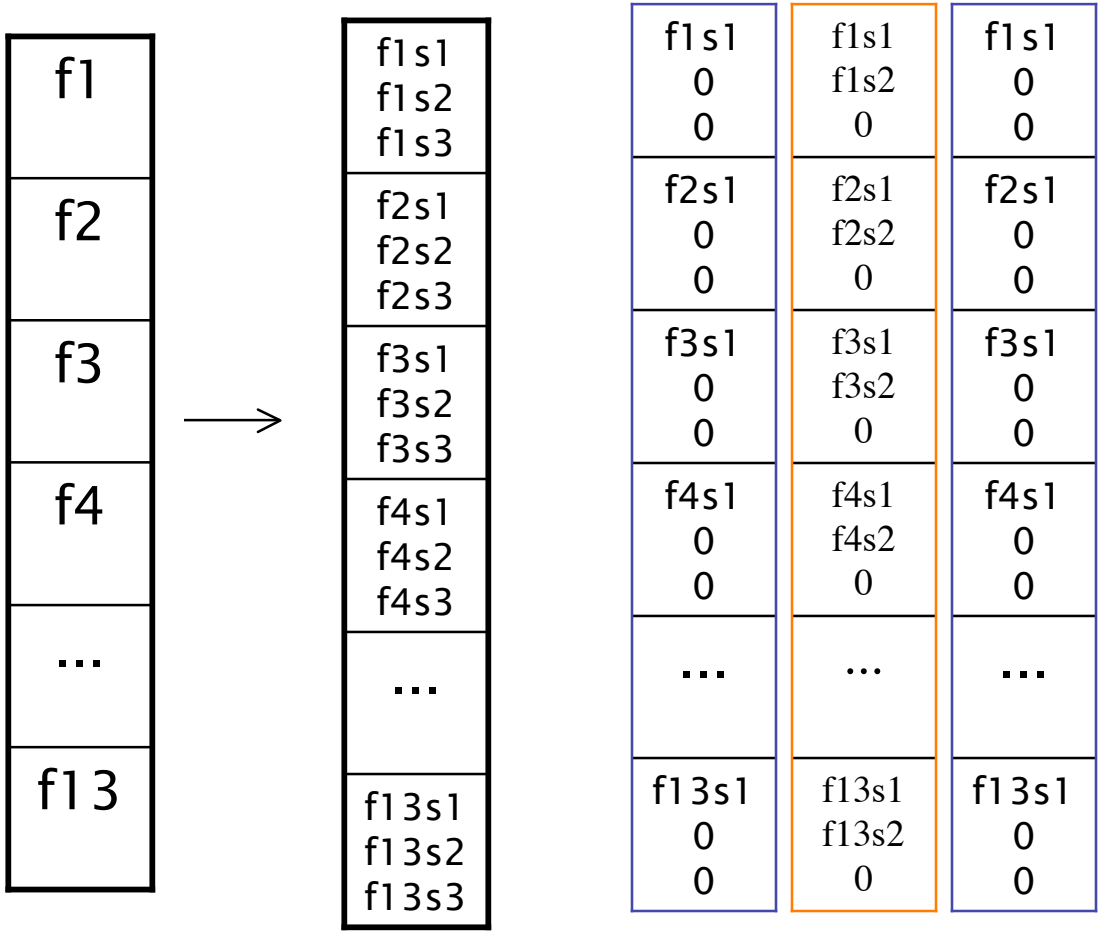
dean rubine: specifying gestures by example
13 unistroke features: 11 geometric, 2 time

dt extractor	rubine extr	description
distfv	f8	total euclidean distance traversed by a stroke
bboxfv2	f3	dimension of bounding box of a stroke
f4fv	f4	stroke bounding box aspect ratio
startendfv	f5	euclidean distance between start and end pts of a stroke
f1fv	f1,f2,f6,f7	sine and cosine of start and end angles of a stroke
f9fv	f9	sum of angle traversed by the stroke
f10fv	f10	sum of absolute angle traversed by the stroke (“curviness”)
f11fv	f11	sum of squared angles traversed by stroke (“jagginess”)
f12fv	f12	max instantaneous velocity within a stroke
f13fv	f13	total time duration of a stroke

generalizing unistroke feature vectors to variable #s of strokes

for fld,svm, nnets, glds..

allocate space for all stroke features in each "feature frame"; preserve frame alignment



perils:

sparse vectors

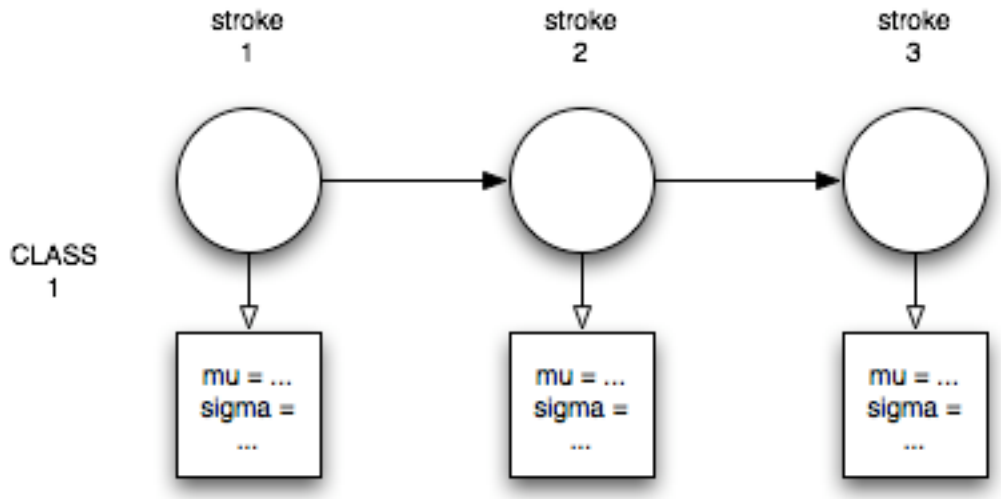
"0" sentinel unintentionally *mis*informative?

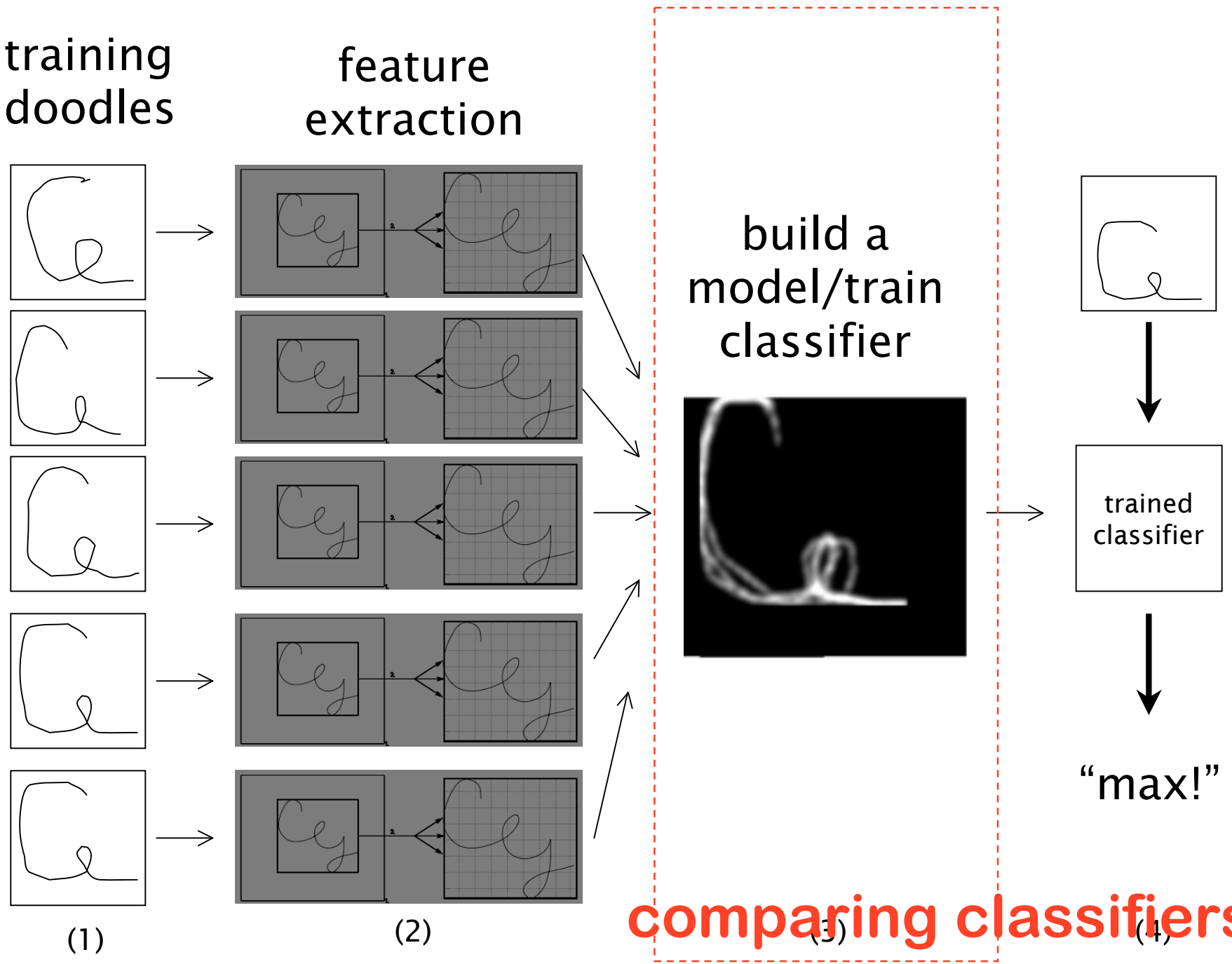
try & see..!

generalizing unistroke feature vectors to variable #s of strokes

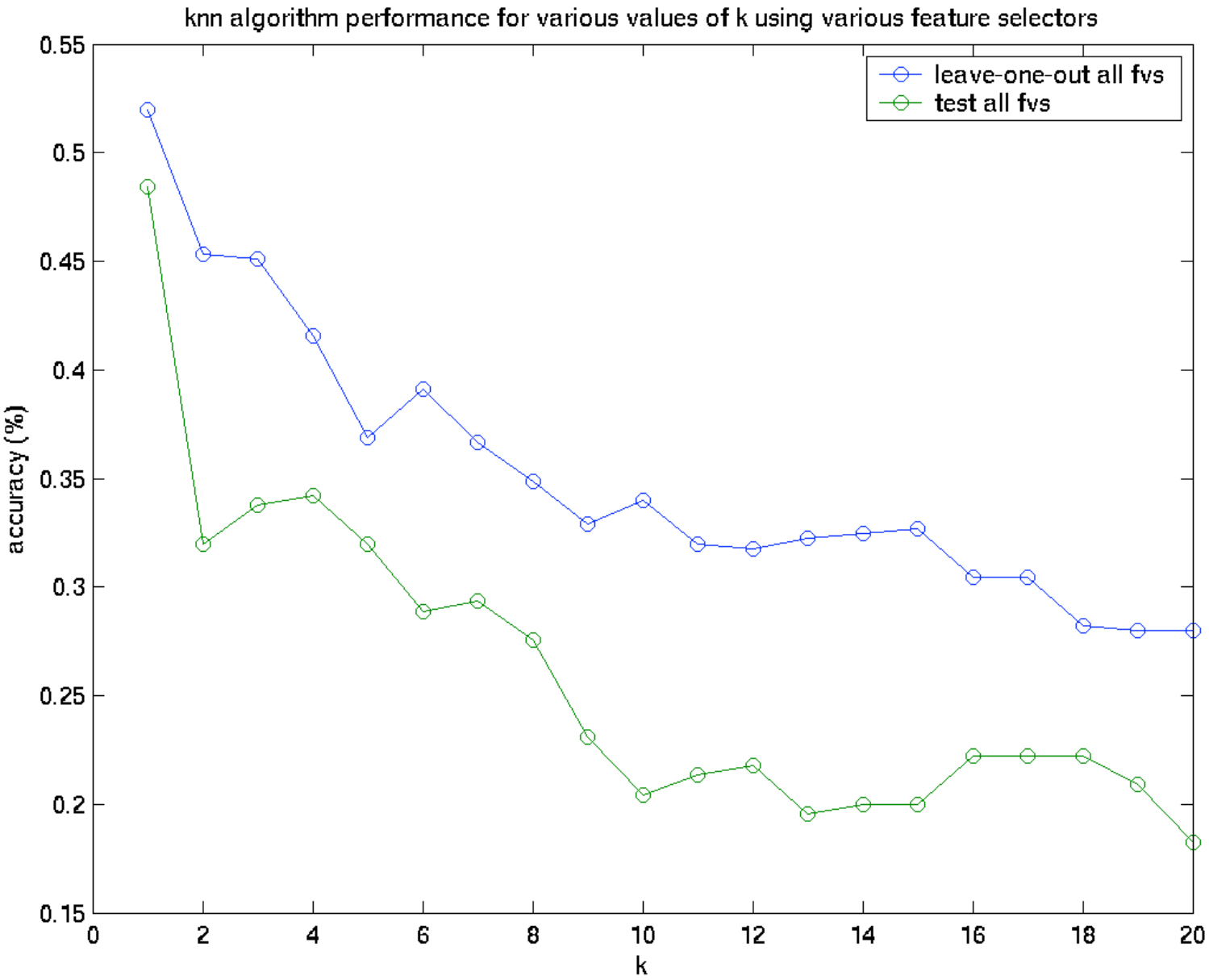
solution #2:

represent each stroke as a sequence; use appropriate techniques

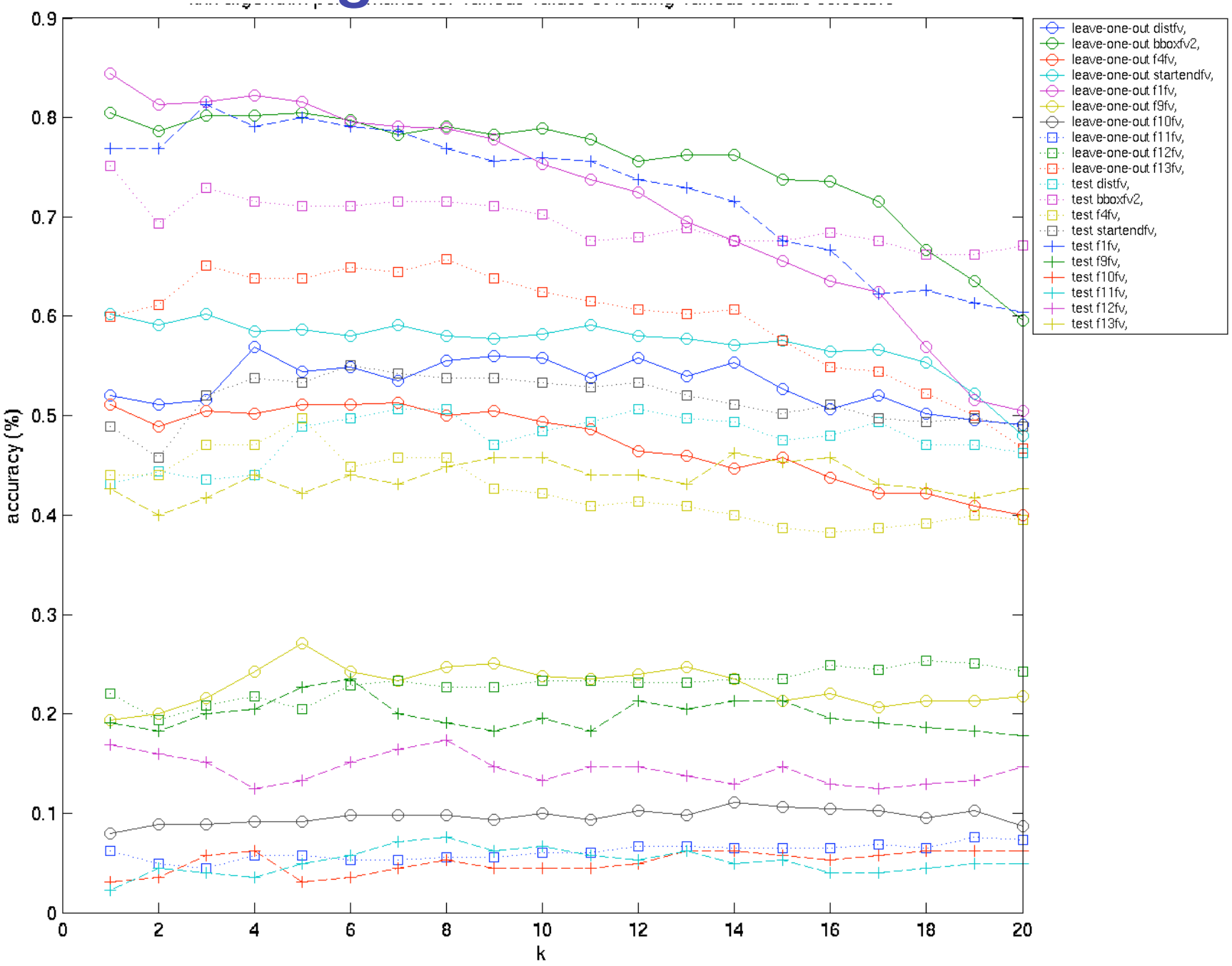



training doodles
feature extraction
build a model/train classifier
trained classifier
"max!"
comparing classifiers
(1)
(2)
(3)
(4)

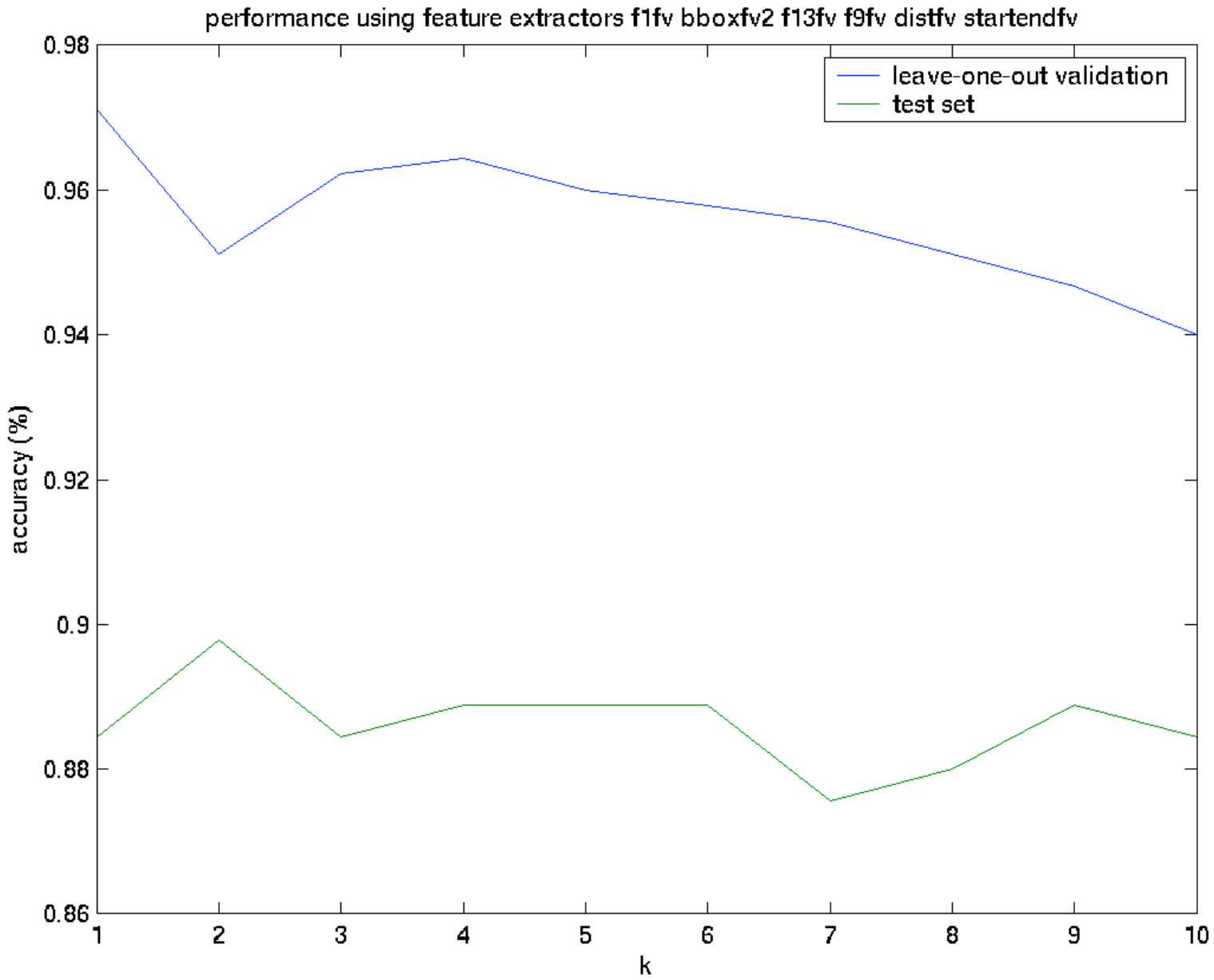
k-nearest neighbors - all features



k-nearest neighbors - individual features



k-nearest neighbors - combinations



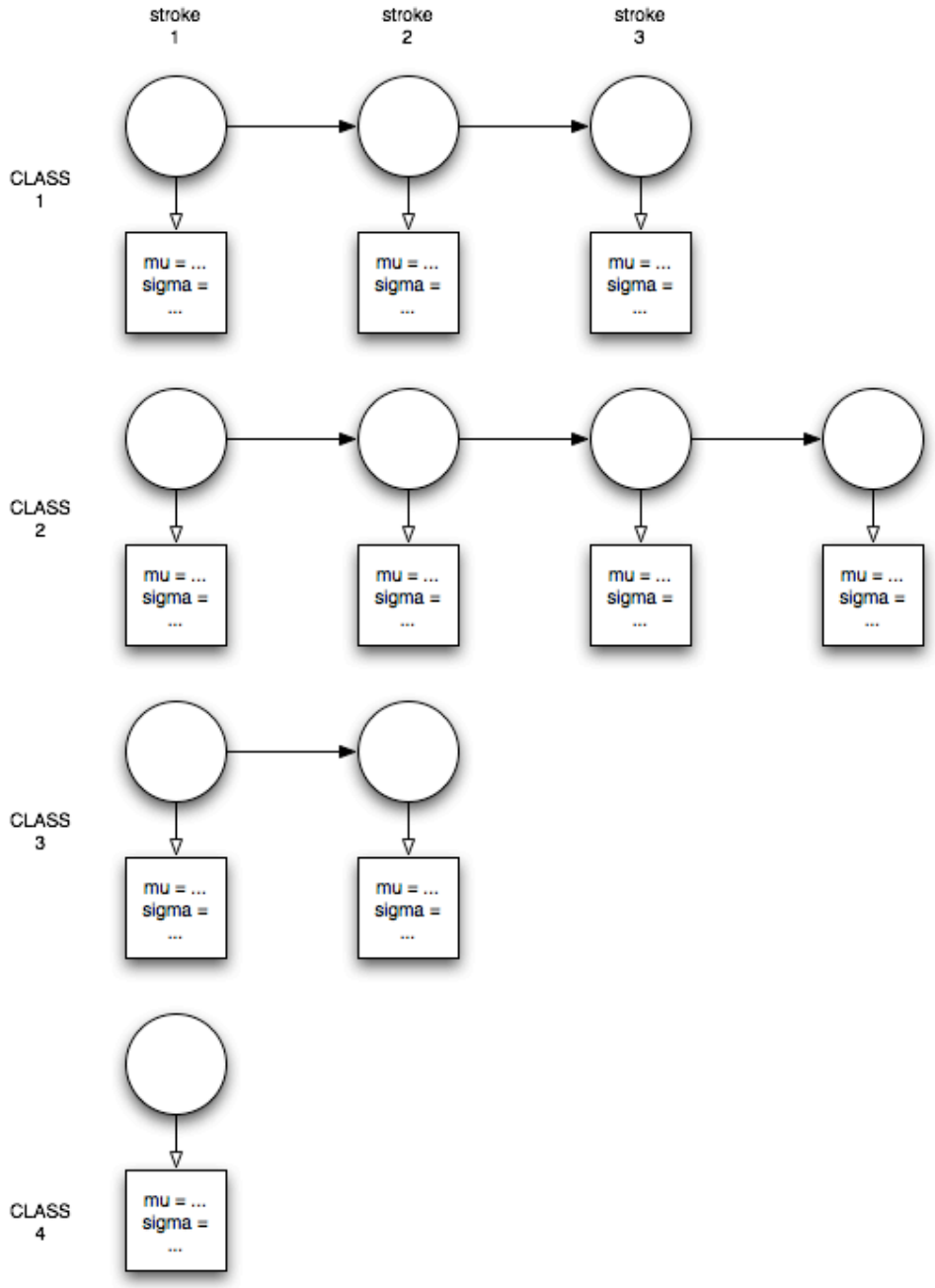
strokewise ML -

simplest multistroke generative model

represent each class as separate sequences of states, each representing a stroke.

Each state thus has an associated parametric distribution over the values of that stroke's feature vector

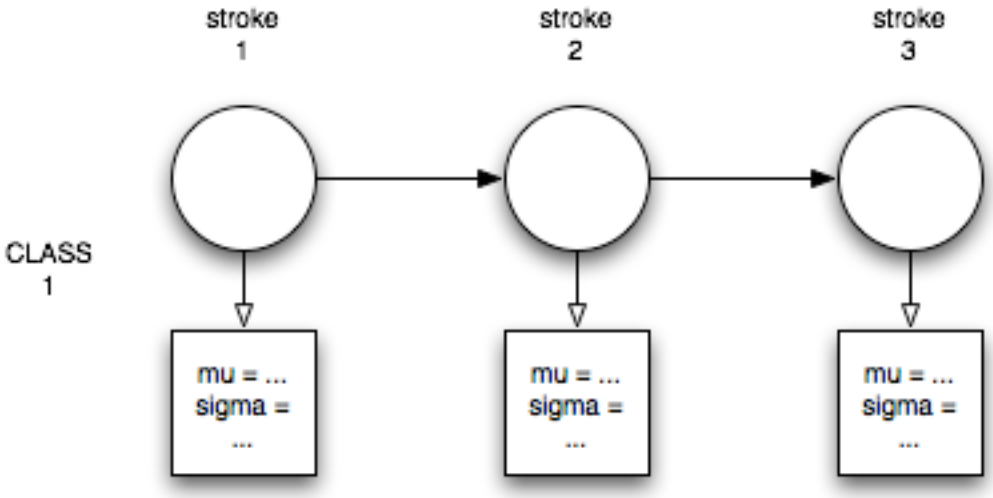
strictly encodes our previous assumption that strokes of the same class always arrive in the same order… otherwise, we'd need an HMM.



strokewise ML - 1 gaussian per stroke

estimation - μ_s Σ_s obtained from ML estimate of
mean, cov of feature vectors for each stroke
in that class

classification - maximum log posterior prob over models
(during comparison, models with different # of strokes
than a gesture are immediately rejected)



easily generalizable to gaussian mixtures - just apply **EM**

strokewise ML - 1 gaussian per stroke

performance with individual features

features	l-o-o performance	test performance
distfv	0.5422	0.3467
bboxfv2	0.8067	0.6756
f4fv	0.5156	0.4267
startendfv	0.6067	0.4400
f1fv	0.8889	0.7511
f9fv	0.6511	0.5822
f10fv	0.5467	0.4222
f11fv	0.4311	0.3333
f12fv	0.2756	0.1289
f13fv	0.6378	0.4178

strokewise ML - 1 gaussian per stroke

performance with multiple features

features	l-o-o performance	test performance
f9fv, bboxfv2, startendfv	0.9733	0.9156
f1fv, startendfv, f9fv, distfv	0.9644	0.8889
f1fv, f9fv, distfv, startendfv	0.9778	0.9644
all combined: distfv, bboxfv2, f4fv, startendfv, f1fv, f9fv, f10fv, f12fv, f13fv	0.9711	0.8800

fisher linear discriminant - (1-dimensional)

OVA (one-versus-all):
train C FLD binary classifiers on the fvs
evaluate each one on the test point
+1 if it gets the label right, 0 otherwise / (C*N)

features	l-o-o performance	test performance
distfv	0.9422	0.8311
bboxfv2	0.9356	0.8711
f4fv	0.8444	0.7556
startendfv	0.9600	0.880
f1fv	0.9244	0.9022
f9fv	0.8711	0.7644
f10fv	0.8467	0.5778
f11fv	0.7867	0.5556
f12fv	0.8467	0.7067
f13fv	0.9333	0.8400

fisher linear discriminant -

combined features
(**warning:** figures are a bit misleading; we'll describe why in the next section)

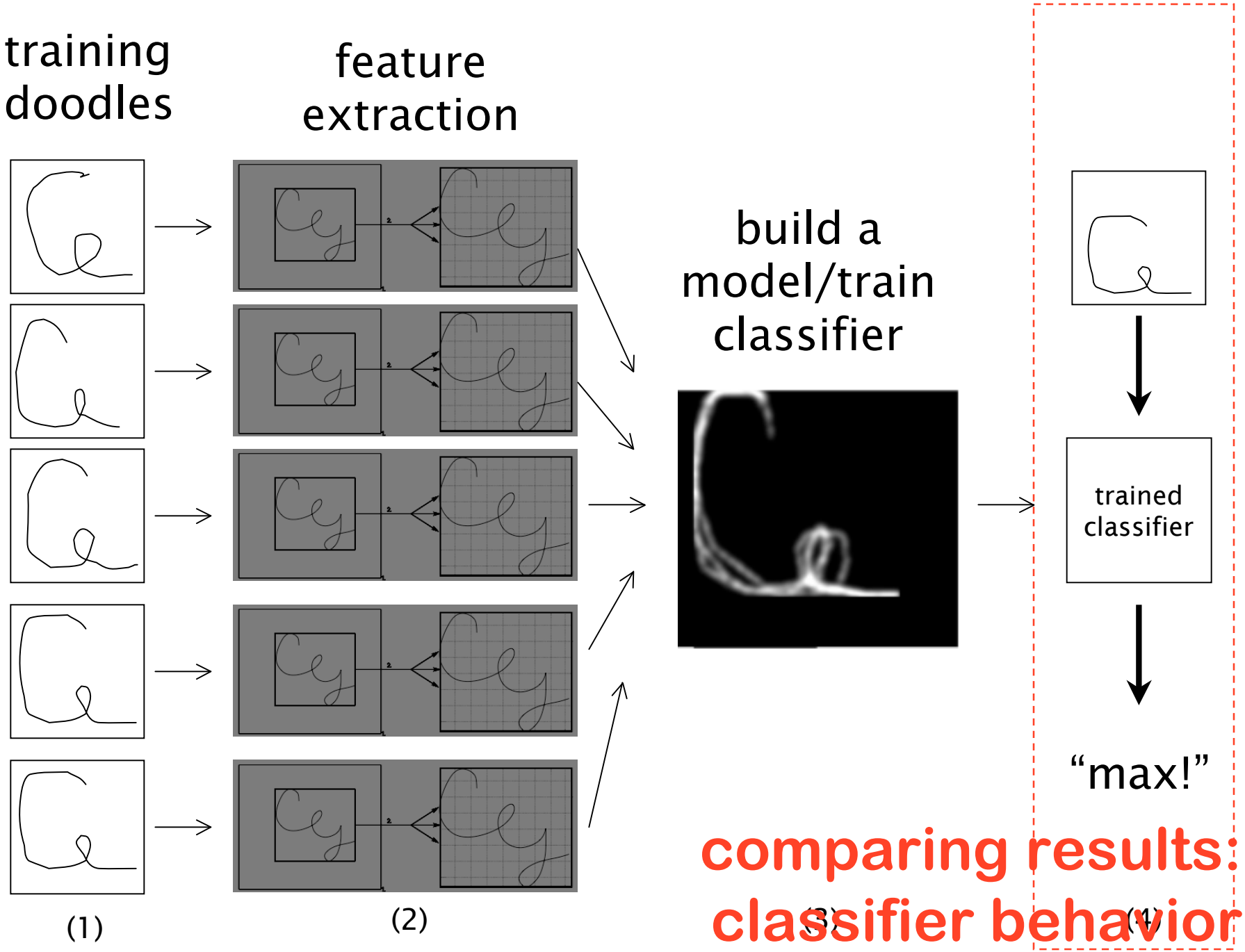
features	l-o-o performance	test performance
bbox2, f12fv, startendfv	0.9822	0.8489
f1fv, f9fv, startendfv	0.9533	0.9733
f1fv, f12fv, distfv, startendfv, bboxfv2	0.9867	0.9156
f1fv, f9fv, f11fv, distfv, startendfv	0.9778	0.9644

support vector machines -

OSU SVM Toolkit for Matlab [http://www.ece.osu.edu/~maj/osu_svm/]

training took too long - no l-o-o

features	linear k test perf.	quad k test perf
distfv	0.9790	0.9790
bboxfv2	0.9789	0.9789
f4fv	0.9784	0.9784
startendfv	0.9778	0.9778
f1fv	0.9831	0.9831
f9fv	0.9778	0.9778
f10fv	0.9778	0.9778
f11fv	0.9778	0.9778
f12fv	0.9774	**0.9775**
f13fv	0.9778	0.9778
all combined	0.9923	0.9923


training doodles
feature extraction
build a model/train classifier
trained classifier
"max!"
comparing results: classifier behavior
(1)
(2)
(3)
(4)

comparison

k-nearest-neighbors - simple, most sensitive to choice of feature extractors

sequential ML - simple to estimate, strictly requires stroke order

fisher linear discriminant (1d) - performed well

support vector machines (lin, quad kernel) - outperformed other methods, took significant training time

rejection

knn - greedily chooses k nearest neighbors
strokewise ML - chooses largest log likelihood

⇒ choose thresholds empirically using l-o-o validation (in theory, tricky in practice - soft thresholds difficult to manage)

FLD and SVMs - gauge 'specificity' of discriminants by measuring performance as follows:

+1 iff all C FLDs/SVMs are correct
0 otherwise
=> "strict criterion"

support vector machines - strict criterion

test performance with polynomial kernel:

features	linear k	quad k	cubic k	quartic k
all features	0.6978	0.6222		
f1fv bboxfv2 distfv, f9fv startendfv	0.7244	0.8444	0.8311	0.7867
f1fv, f9fv, f11fv, bboxfv2, distfv	0.6489	0.8400	0.8000	0.7511

comments?

http://people.csail.mit.edu/~emax/dt
write me: max@mit.edu

(by the way, matlab runs much faster +
crashes less without the GUI!
matlab -nodesktop)

..good night!