

Chapter 1

Handwriting Recognition Engine

1.0.1 Curve Handling

In this prototype the direction feature plays a crucial role. Generally following several approaches in on-line handwriting recognition literature (see section ??), the approach used here defines direction vectors. The general approach is to define a direction vector between a point within the point sequence and a successor point. However, since the measured points are usually very close together, the direction vector is defined between a point and another point that is further away, defined by a dynamic threshold, considering distance, total number of points and the total length of the stroke.

If non of these vectors deviates from the general direction of the stroke, above another dynamic threshold, considering the total stroke length, the stroke is considered straight. Nevertheless, for matching straight strokes, a feature vector is used.

Different curve matching techniques are described in section ?. The technique used here refers to *curved strokes*, however, it is a vector-based technique, that aims at point reduction.

S 14, 16, 17 how is curver handling done? show requirements. what alternatives were there to consider? stroke matching with angles instead of point position. s. 24

1.0.2 Dynamic Time Warping

what's the similarity measure for points and strokes? show requirements. what alternatives were there to consider?

s. 51 how is dynamic time warping done here? pointer to papers or hwr - chapter, don't explain DTW here. show requirements why DTW? what alternatives were there to consider? none - it is the alternative. to all the other stuff I've been doing. however, what about 3D time warping?

1.1 Radical Recognition Process

1.2 Character Recognition Process

In order to scale to the normalised size, the length of an edge of the bounding box is compared to the length of the normalised character.

1.3 Error Handling

see section ?? in chapter ?? for possible sources of error

1.3.1 Error Recognition

why this section? to demonstrate own achievements of error recognition. the reader should know how it is done technically.

what goes into this section? the aspects of finding errors. finding errors is not a straightforward trivial task - whenever something does not match it is an error - doesn't work like that. instead, firstly, it needs to be made sure that it actually is an error. meaning - not a recognition error, but a user error. secondly, the type of error needs be identified. see section ?? (or handwritten page 58) for sources of error.

how will this section be written? technical - first describe how the error recognition integrates into the recognition process, then how errors are identified.

1.3.2 Error Processing

why this section? actually the 'handling' or 'processing' aspect could be described in the recognition section 1.3.1 as well. so this section is only for a better overview, for document structure, thematically they are the same section. thus they are put together under Error Handling 1.3.

what goes into this section?