



ON THE SEGMENTATION AND CLASSIFICATION OF HAND RADIOGRAPHS

LUKE M. DAVIS*, BARRY-JOHN THEOBALD[†] and JASON LINES[‡]

*School of Computing Sciences, University of East Anglia
Norwich Research Park, Norwich, Norfolk
NR4 7TJ, United Kingdom*

**luke.davis@uea.ac.uk*

†b.theobald@uea.ac.uk

‡j.lines@uea.ac.uk

ANDONI TOMS

*Norwich Radiology Academy
Norfolk and Norwich University Hospital
Norwich, Norfolk, NR4 7UB, United Kingdom
andoni.toms@nnuh.nhs.uk*

ANTHONY BAGNALL

*School of Computing Sciences, University of East Anglia
Norwich Research Park, Norwich, Norfolk
NR4 7TJ, United Kingdom
anthony.bagnall@uea.ac.uk*

Accepted 8 July 2012

Published Online 23 August 2012

This research is part of a wider project to build predictive models of bone age using hand radiograph images. We examine ways of finding the outline of a hand from an X-ray as the first stage in segmenting the image into constituent bones. We assess a variety of algorithms including contouring, which has not previously been used in this context. We introduce a novel ensemble algorithm for combining outlines using two voting schemes, a likelihood ratio test and dynamic time warping (DTW). Our goal is to minimize the human intervention required, hence we investigate alternative ways of training a classifier to determine whether an outline is in fact correct or not. We evaluate outlining and classification on a set of 1370 images. We conclude that ensembling with DTW improves performance of all outlining algorithms, that the contouring algorithm used with the DTW ensemble performs the best of those assessed, and that the most effective classifier of hand outlines assessed is a random forest applied to outlines transformed into principal components.

Keywords: Hand segmentation; outline classification; outline ensemble.

1. Introduction

This research is part of a wider project to build predictive models of bone age using hand radiograph images. Bone age assessment typically involves estimating the expected age of a patient from a radiograph by quantifying the development of the bones of the non-dominant hand. It is used to evaluate whether a child's bones are developing at an

acceptable rate, and to monitor whether certain treatments are affecting a patient's skeletal development. Currently, this task is performed manually by scoring each bone using a system such as Tanner and Whitehouse (TW).¹ This procedure is time consuming and often inaccurate. We are attempting to develop a fully automated system that will quickly produce an accurate estimate of bone age

based on predictive models constructed from a wide range of images. Broadly speaking, bone age is estimated by the shape and position of the bones in the hand. Hence, our approach involves segmenting the image to find the location of individual bones, then extracting features to build a regression model. This approach is also adopted in a recently published algorithm that forms the basis of a commercial piece of bone ageing software called boneXpert.² This software constructs Active Appearance Models (AAM) of the bones directly from the radiographs. Our experience indicates that by first extracting the outline of the hand we will get a better segmentation of the bones as the hand outline gives clearly defined landmarks upon which to begin the extraction. This paper addresses two problems associated with the task of extracting the outline of a hand from an image: Firstly, we assess how to extract the outline, and secondly, we evaluate how to determine whether a given outline is in fact a correct outline of a hand. We assess four candidates for extracting the hand outline: Otsu thresholding,³ Canny edge detection,⁴ AAM,⁵ and contouring.⁶ These are briefly described in Sec. 3. Despite the fact that the first three candidates have previously been used to extract hand outlines,^{7–9} our experience is that the variability in intensity across images, low contrast between the background, flesh, and bone, and variability in hand size and shape mean that the extraction of the outline is non-trivial, and that none of the algorithms assessed are consistent enough for our requirements. Therefore, we define an ensemble method that combines the outlines formed from a range of transformed images (see Sec. 4), which we find creates

much better outlines (as assessed in Sec. 6). However, none of the algorithms we evaluate work perfectly. Figure 1 shows two examples of incorrect outlines. In these examples the algorithm has found the internal outline of the metacarpals or phalanges. We have observed several other types of error, such as over extended regions or cropping of individual fingers.

Clearly, an incorrectly segmented outline will compromise any subsequent steps of bone segmentation and age modeling. Since our aim is to minimize the human intervention needed to progress from image to age estimate, we require an automated means of classifying whether an outline is correct. Section 2 covers the history of bone age assessment and how it is performed currently in hospitals worldwide. We constructed a training set of 1000 images and a test set of 370 images from data collected from the Children’s Hospital Los Angeles.¹⁰ The images are from males and females in the age range 0–18 years. Each image was automatically segmented (Secs. 3 and 4) and manually labeled as correct or incorrect depending on the quality of the segmentation. Features were then extracted from the training images and a range of classification algorithms (Sec. 5) were evaluated using the testing set. The results are presented and analyzed in Sec. 6. In Sec. 7 we conclude from our findings and describe how this prototype system could be improved.

This work is an extension of that performed by Davis *et al.*¹¹ The contributions of this paper can be summarized as follows:

- (1) We propose using contouring to find hand outlines in radiographs, and compare contouring

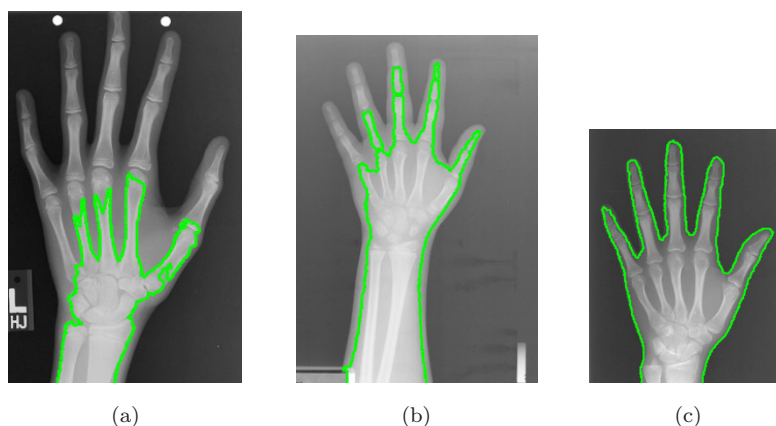


Fig. 1. (a) and (b) Two examples of incorrectly located hand outlines, and (c) a hand outline correctly segmented.

with other methods used in this problem domain. To our knowledge, contouring has not been used in this context before and our experiments indicate that it has great potential.

- (2) We propose a new way of combining image outlines through an ensemble technique, with diversity achieved through rescaling and two novel voting scheme that use a likelihood ratio test and dynamic time warping.
- (3) We describe a classification problem based on a one-dimensional transformation of the hand outline and evaluate a range of alternative classifiers on different transformations of the outline.
- (4) We demonstrate that ensembling with dynamic time warping using contouring produces the most reliable outlines.

2. Bone Age Assessment

Bone age measures the current state of skeletal maturity against the normal development for a given population. Difference in bone age and chronological age can indicate medical problems.^{12,13} In clinical radiology, bone age assessment is a task regularly performed by pediatricians to monitor skeletal development and the effects of certain drugs.¹⁴

2.1. Manual bone age assessment

Bone age assessment is currently performed in hospitals worldwide on a daily basis. This procedure is undertaken by obtaining a radiograph of the patient's nondominant hand. It is widely accepted that the hand is a good indicator of skeletal maturity.^{1,15-17} This is for three main reasons. Firstly, it has many ossification centers in a small area. Secondly, due to the small area, the patient is exposed to minimal radiation. Finally, it is an easy area to radiograph. An examination of the skeletal development of the hand is undertaken using one of two methods: The Atlas method of Greulich and Pyle^{18,19} or the scoring method proposed by Tanner and Whitehouse.^{1,20,21}

2.1.1. The Greulich and Pyle method

The Greulich and Pyle atlas¹⁹ consists of a set of reference images over a range of ages. There are 31 reference images for males and 26 for females. The clinician checks a patient's radiograph against each of

the example radiographs in the atlas to find the most similar. When comparing against each radiograph, certain features of the skeletal development, such as the development of the epiphysis and the presence of certain carpal bones, should be checked. The bone age prediction is that of the image in the atlas deemed closest to that of the patient in question.

The main disadvantages of using this method are that it is subjective and therefore harder to reproduce diagnoses.^{22,23} The use of an atlas method assumes that the ossification process happens in an orderly fashion among all people; however, this may not be the case.²² Another criticism is the long intervals between standards. A study in 1960s Denmark found that the standards of the atlas were not a good representation of their population,²⁴ although a more recent study was performed in the Netherlands and found the atlas to still be of use.²⁵

2.1.2. The Tanner and Whitehouse method

In contrast to the atlas based method of Greulich and Pyle,^{18,19} the Tanner and Whitehouse method^{1,20,21} grades a selection of bones. Each bone has various stages of development, and each stage has various criteria for a bone to be scored at that stage. Bone scores are summed to give the skeletal maturity score. This is converted into a bone age using a chart.

The advantages of using the Tanner and Whitehouse method, as opposed to the Greulich and Pyle method, are that it overcomes the subjectivity problem and results are more reproducible.²³ However, rating individual bones is time-consuming, and so the Greulich and Pyle method is used more often.

2.2. Automated bone age assessment

Automated bone age prediction requires image segmentation, feature extraction and classification/regression.

Pietka *et al.*²⁶ describe a method that uses c-means clustering and Gibbs Random Fields to extract the bones from a radiograph. Six regions of interest (ROI) are located: The joints between distal phalanges — middle phalanges and middle phalanges — proximal phalanges of fingers two, three and four. These ROIs are segmented and features are extracted from them. Firstly, background subtraction is performed using histogram analysis, and

thresholding is performed to extract the hand. The axes of fingers two, three and four are then located. These axes are used to locate the six ROIs. The bone is extracted from the soft tissue in the ROIs.

Thodberg *et al.*² use AAMs to assess skeletal maturity in their software BoneXpert. This algorithm consists of three layers: A, B and C. AAMs are used in layer A to reconstruct bones and determine how “normal” they are. The authors annotated 3000 hand radiographs by hand to model the 15 RUS bones (Radius Ulna and Short Bones). The bone age is constructed in layer B, but does not use the GP or TW method, instead it uses the shape and intensities learnt from the AAMs. The BoneXpert bone age is then converted into GP or TW bone age in layer C, the only time in the process that human ratings are used.

Our objective is to develop an approach that is fully automated and can be adapted to local populations. Clearly, the image segmentation stage presents a major challenge. We assess variants of the approaches taken by Pietka and Thodberg in the context of hand outlining.

3. Outlining a Hand

There has been extensive research on the segmentation of hand radiographs.^{7–9,27} The majority of this work concentrates on the direct segmentation of the bones and uses the problem of finding the outline merely as a motivational example. However, we consider the seemingly easier problem of finding the hand outline as the most sensible first step; once we have obtained an outline that we are confident is correct, the position of the bones is highly constrained and thus much easier to detect. We have applied four commonly used algorithms for outlining, described below.

3.1. Active appearance models

Active shape models (ASM)^{28–30} and Active Appearance Models (AAM)^{2,9,31} have been used previously for the segmentation of bones from radiographs. However, the use of ASMs for this task has decreased since the introduction of AAMs, as the AAM uses a more complete model of an image. To use AAMs to extract the hand outline, the model must first be trained over a set of manually annotated images. This is performed by placing k landmarks along the

outline of the hand. The landmarks are then normalized for translation, rotation and scale and principal components analysis (PCA) is applied, to give a compact model of shape of the form:

$$\mathbf{s} = \bar{\mathbf{s}} + \mathbf{P}\mathbf{b}_s, \quad (1)$$

where \mathbf{s} refers to a set of landmarks in an image, $\bar{\mathbf{s}}$ is the mean shape, \mathbf{P} is the set of eigenvectors that define the allowed variation of the shape, and \mathbf{b}_s are the shape parameters.

AAMs also encompass a model of the intensity variation within the shape. The labeled training images used to construct the shape model are warped to the mean shape, $\bar{\mathbf{s}}$, and PCA is applied to these shape normalized images. This provides a compact model of appearance variation of the form:

$$\mathbf{a} = \bar{\mathbf{a}} + \mathbf{P}\mathbf{b}_a, \quad (2)$$

where \mathbf{a} refers to a shape normalized image, $\bar{\mathbf{a}}$ is the mean appearance, \mathbf{P} is the set of eigenvectors that define the allowed variation of appearance, and \mathbf{b}_a are the appearance parameters.

Once the model has been built, a further iterative algorithm is used to fit the outline to new instances. We have used the Inverse Compositional AAM.³²

AAMs are a powerful method for tasks where the user wants to classify objects by shape. However, the object being modeled needs to be well defined. A large set of example images of the object are required for a large amount of variation to be modeled.

Some of the advantages of this method are as follows. The model can incorporate the knowledge of an expert from the annotation of the training examples e.g. knowing the difference between a bone in Stage E and a bone in Stage F of the TW method. AAMs are able to model the variation of shape and texture in a compact representation and only needs the knowledge of the object gained from the training set.

3.2. Otsu thresholding

The Otsu method³ for thresholding hand radiographs^{7,26} uses the probability distribution, \mathbf{p} , of the pixel intensities of the input hand radiograph. The mean pixel intensity, μ_T , is calculated using:

$$\mu_T = \sum_{j=0}^{255} j p_j. \quad (3)$$

A pixel intensity, i , is classified as background with a given probability, $\omega(i)$, using the cumulative probability distribution:

$$\omega(i) = \sum_{j=0}^i p_j. \quad (4)$$

The mean intensity of pixels up to level i , $\mu(i)$, and the between class variance of intensities $\sigma_B^2(i)$, are calculated using Eqs. (5) and (6), respectively.

$$\mu(i) = \sum_{j=0}^i j p_j \quad (5)$$

$$\sigma_B^2(i) = \frac{[\mu_T \omega(i) - \mu(i)]^2}{\omega(i)[1 - \omega(i)]}. \quad (6)$$

The optimal threshold, $\sigma_B^2(i^*)$, is calculated using:

$$\sigma_B^2(i^*) = \max_{0 \leq i \leq 255} \sigma_B^2(i), \quad (7)$$

which provides the division of the radiograph into background and foreground (hand) regions.

Unlike the AAM method, the Otsu method presents a fully automated solution. The advantage of such a system is that it avoids the need for a person to label potentially hundreds of images.

However, the disadvantages of such a method are that it does not contain any shape and/or appearance information. This could mean that the foreground selected using this method may not be the desired object, due to an unknown artifact affecting the probability distribution of the image. There may be more than one outline in the binary mask after thresholding. We assume the largest object in the radiograph to be the hand and extract the outline from it.

3.3. Canny edge detector

The Canny edge detector⁴ is a multistage algorithm that combines differential filtering, nonmaximal filtering, and thresholding with hysteresis. It has been used previously in the context of hand radiograph segmentation.^{8,33} To summarize, the Canny algorithm:

- Smooths the image with a Gaussian filter.
- Estimates the gradient magnitude and direction at each pixel and quantizes the gradient directions to be one of $\{0, 45, 90, 135\}$ degrees.

- Performs nonmaximal suppression by switching off candidate pixels that are not locally maximum in the direction of the gradient.
- Identifies definite edge pixels as those with a gradient magnitude above a global high value threshold, and switches off pixels that have a gradient magnitude below a global low threshold.
- Checks the pixels with gradient magnitude between the high and low thresholds to determine if there is a path that connects them to a definite edge pixel. Those that are connected to a definite edge form the edge, otherwise they do not.

As with the Otsu method, the Canny edge detector is a fully automated solution to image segmentation, requiring no hand annotation of images.

A major disadvantage of using this method is that it detects edges, not necessarily outlines. This could cause a problem if there are no strong edges around the object to be segmented. The Canny edge detector identifies all edges in an image, and we assume that the longest edge represents the hand outline.

3.4. Contour algorithms

To the best of our knowledge, contour algorithms have not been used in this context, but have been used previously for tasks such as weather analysis,⁶ gesture recognition,³⁴ and road sign recognition.³⁵ The contour algorithm used takes an input radiograph \mathbf{I} with intensity range 0–255. It calculates n contour levels c_1, c_2, \dots, c_n where $c_1 < c_2 < \dots < c_n$. The contour levels are at n equally spaced points between the minimum and maximum pixel intensities. A simple example input can be seen in Fig. 2, which would have contour levels at 50, 100 and 150. For each contour level c_i , pixels that have an edge intersecting c_i are found. In Fig. 3 we show the edges in our example at $c_i = 50$.

0	75	0
125	200	160
75	125	155

Fig. 2. A simple example input image \mathbf{I} .

0	75	0
125	200	160
75	125	155

Fig. 3. The example shown in Fig. 2 with edges at 50 highlighted using a bold line.

0	75	0
125	200	160
75	125	155

Fig. 4. The example shown in Fig. 2 with the first edge at 50 highlighted.

The first edge is found and highlighted (Fig. 4). The t intercept and contour point (p_x, p_y) are calculated using Eqs. (8), (9) and (10) respectively. c_i is the contour level, z_0 and z_1 are the pixel intensities either side of the edge, (x_0, y_0) and (x_1, y_1) are the co-ordinates respectively of the pixels. The t intercept refers to the point between the two pixels where the contour level intercepts. A visualization of this is shown in Fig. 5.

$$t = \frac{c_i - z_0}{z_1 - z_0} \tag{8}$$

$$p_x = x_0 + (t(x_1 - x_0)) \tag{9}$$

$$p_y = y_0 + (t(y_1 - y_0)). \tag{10}$$

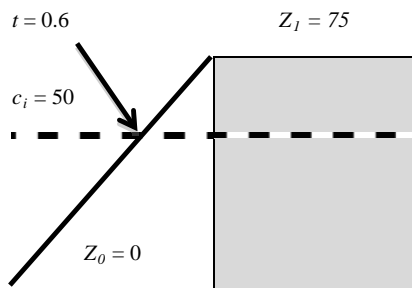


Fig. 5. An example of the t value calculation.

0	75	0
125	200	160
75	125	155

Fig. 6. The example shown in Fig. 2 with the first edge at $c_i = 50$ recorded (bold line) and the connecting edges being shown (dashed line).

Based on our example the calculations would be:

$$t = \frac{50 - 0}{75 - 0} = 0.6 \tag{11}$$

$$p_x = 1 + 0.6 \times (2 - 1) = 1.6 \tag{12}$$

$$p_y = 1 + 0.6 \times (1 - 1) = 1. \tag{13}$$

The contour point (p_x, p_y) is recorded as the first point of the contour \mathbf{C} . The edges connecting to the current edge are checked to see if they intersect level c_i , shown in Fig. 6.

If there is a connecting edge that intersects level c_i as shown in Fig. 7, the process is repeated with the t intercept, contour point \mathbf{p} being calculated (see Eqs. (14) to (16)), and \mathbf{p} being concatenated onto the contour \mathbf{C} (Fig. 8).

$$t = \frac{50 - 0}{125 - 0} = 0.4 \tag{14}$$

$$p_x = 1 + 0.4 \times (1 - 1) = 1 \tag{15}$$

$$p_y = 1 + 0.4 \times (2 - 1) = 1.4. \tag{16}$$

The contour \mathbf{C} is terminated when any of the following happen:

- All connecting edges are not intercepted by c_i .
- The contour returns to an edge it has already visited.

0	75	0
125	175	160
75	125	155

Fig. 7. The example shown in Fig. 2 with the first edge at $c_i = 50$ recorded (bold line) and the connecting edge where the contour level intercepts (dashed line).

0	75	0
125	200	160
75	125	155

Fig. 8. The example shown in Fig. 2 with the edges recorded in \mathbf{C} at $c_i = 50$ are highlighted (bold line).

- The contour leaves image \mathbf{I} (as in the example shown).
- All connecting edges have been marked by other contours.

After each contour \mathbf{C} at each contour level c_i has been found, the set of contours \mathbf{S} is returned.

The output from the contouring algorithm is a set of contours, and we take the largest contour to represent the outline of the hand.

4. Ensemble Techniques

There are two main factors that make finding a hand outline difficult. Firstly, the background/hand division we are attempting to find can be obscured by the hand/bone division, which is often more pronounced. Secondly, the distributions of pixel intensities vary greatly from image to image. This is caused by differences in the machine used, deterioration of the bulb over time, and the fact that the energy emitted from the bulb is nonuniform across the bulb (this is commonly referred to as the Heel Effect; an example is shown in Fig. 9). One way of overcoming the first problem is to rescale an image \mathbf{I} to extenuate



Fig. 9. An example radiograph displaying the heel effect.

the background/hand division using a power transform, $\mathbf{I}^\gamma + c$ (where c is a constant that rescales the intensities back in the range 0–255). However, the second problem of variation in the distribution of intensities means the optimal γ value is image dependent. Hence we propose an ensemble approach. This involves creating twenty rescaled images with scaling factor $\langle \gamma_1 = 0.1, \gamma_2 = 0.2, \dots, \gamma_{20} = 2.0 \rangle$. Each rescaled image is independently outlined by one of the algorithms described in Sec. 3. Each outline O_i is a list of (x, y) co-ordinates derived from the image transformed with power γ_i .

Once the 20 outlines have been created, the problem is to choose one. Unlike traditional classification ensembles,^{36,37} we cannot simply vote on an outline. Instead, we propose two separate selection methods. The first method is based on comparing the shape of the outline to a set of idealized outlines (Sec. 4.1). The second involves using a test statistic on the difference in intensity distributions of the image inside and outside the outline (Sec. 4.2).

4.1. Dynamic time warping outline selection

Dynamic time warping (DTW) is an elastic measure of the similarity of 1-D series that has become popular in time series data mining.³⁸ Suppose $M(Q, C)$ is the $n \times m$ point-wise Euclidean distance matrix between two ordered series $Q = \langle q_1, \dots, q_n \rangle$ and $C = \langle c_1, \dots, c_m \rangle$, where $M_{i,j} = (q_i - c_j)^2$. A warping path $W = \langle (a_1, b_1), (a_2, b_2), \dots, (a_k, b_k) \rangle$ is a set of index pairs that define a traversal of matrix M . A valid warping path must satisfy the conditions $(a_1, b_1) = (1, 1)$ and $(a_k, b_k) = (n, m)$ and that $0 \leq a_{k+1} - a_k \leq 1$ for all $k < n$ and $0 \leq b_{k+1} - b_k \leq 1$ for all $k < m$. The distance for any path W is:

$$D_W(Q, C) = \sum_{i=1}^k M(a_i, b_i). \quad (17)$$

The DTW distance between series is the path through M with the minimum total distance. Let \mathbf{W} be the space of all feasible paths. The DTW path W^* is the path that has the minimum distance, i.e.

$$W^* = \min_{W \in \mathbf{W}} (D_w(Q, C)). \quad (18)$$

To apply DTW to hand outlines we first map each outline onto a 1-D series by computing the Euclidean distance of each pixel along the contour

to the midpoint of the wrist. Hence for any outline $O = \langle (x_i, y_i), \dots, (x_n, y_n) \rangle$, the associated 1-D series is defined as:

$$Q = \langle q_i = (x_m - x_i)^2 + (y_m - y_i)^2 | 1 \leq i \leq n \rangle. \quad (19)$$

Where

$$x_m = \frac{x_n - x_1}{2} \quad (20)$$

$$y_m = \frac{y_n - y_1}{2}. \quad (21)$$

Illustrative examples of converting good and bad outlines into a 1-D series are shown in Figs. 10 and 11, respectively. These show the difference between 1-D series and hence why we believe DTW would be applicable for this task. For any given image, we create 20 candidate series from our outlines. We wish to select the series that most resembles a correct hand outline using DTW to measure similarity. Of course, there is a wide variation in possible correct outlines. In order to get a range of ground truth candidates we took the 59 idealized

radiographs that are presented in Ref. 39. These range in age from 8 months to 18 years. We manually outlined these images to form a set of correct outlines, C_1, \dots, C_{59} . Our selected outline is the outline O_k that has the minimum median DTW distance to our set of correct outlines, i.e. $\min DTW(O_k, C_j)$. An example of DTW between two 1-D series of hand outlines is shown in Fig. 12.

4.2. Likelihood ratio outline selection

An alternative approach to DTW is to select an outline based on the intensity distribution of the original image both inside and outside of the outline. Given an outline O , let the set of points in the original image inside the outline be A and the set of points outside the outline be B and let $|A| = n_a$ and $|B| = n_b$. An image has intensity values in the range 0 to 255. Let the number of points in set A with intensity k be a_k and the number of points in set B with intensity k be b_k . The counts a_k and b_k form the histograms of intensity occurrences inside

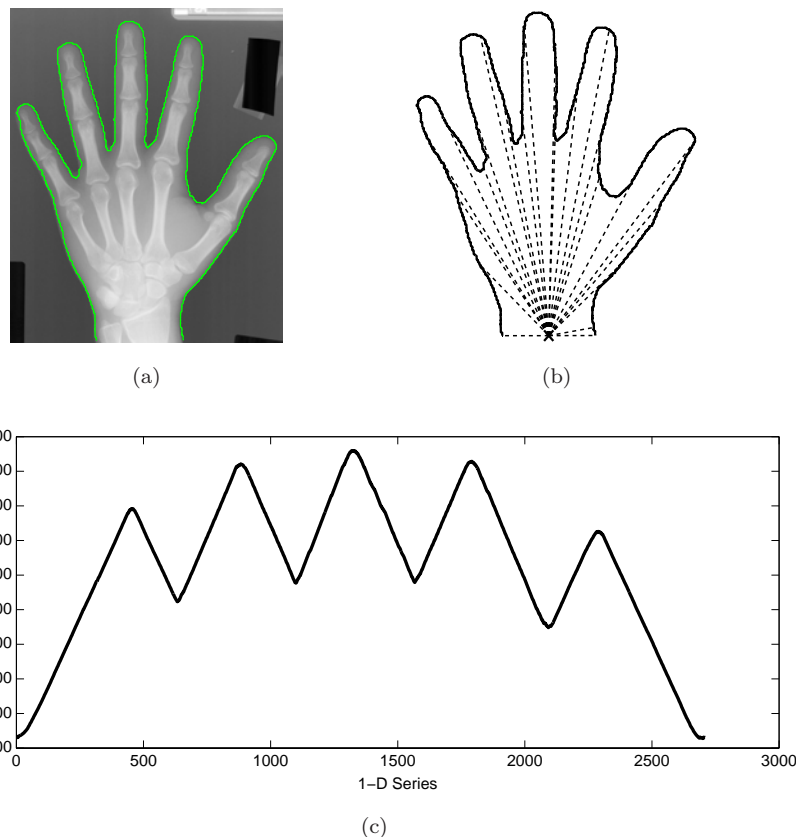


Fig. 10. An example of a good hand outline from a radiograph (a) being converted into a 1-D series (b) and (c).

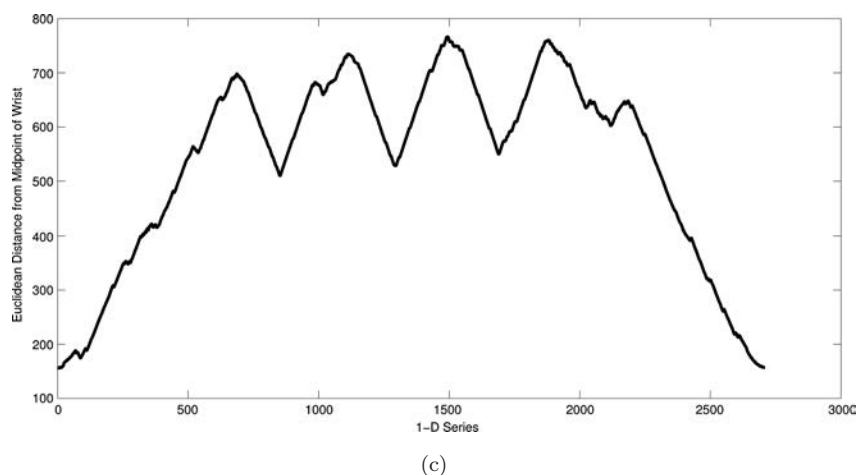
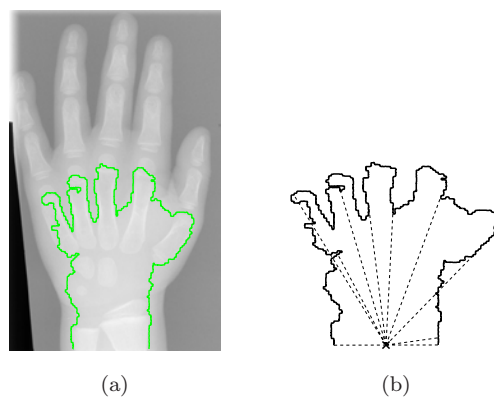


Fig. 11. An example of a bad hand outline from a radiograph (a) being converted into a 1-D series (b) and (c).

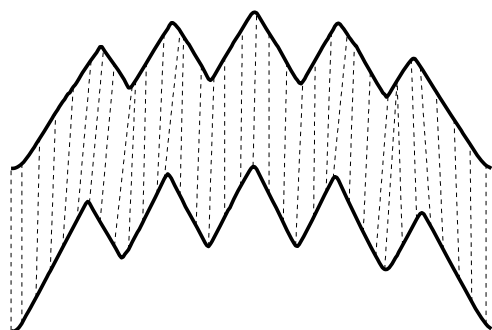


Fig. 12. An example of DTW between two 1-D series of hand outlines.

and outside of the outline. We can form the intensity distributions from the relative frequencies,

$$p_{a_k} = \frac{a_k}{n_a}, \quad p_{b_k} = \frac{b_k}{n_b}. \quad (22)$$

We wish to choose the outline where the distribution within the outline is most different from that outside of the outline. To do this we use the likelihood ratio statistic for the test of the null hypothesis

that the distributions are equal. Under this null, our probability estimates are:

$$p_k = \frac{a_k + b_k}{n_a + n_b}, \quad (23)$$

and the test statistic is given by the log of the likelihood ratio,

$$\begin{aligned} d_L(O, I) &= \log(L(A, B)) \\ &= - \sum_{k=0}^K p_{a_i} \log \left(\frac{p_{a_i}}{p_i} \right) + p_{b_i} \log \left(\frac{p_{b_i}}{p_i} \right), \end{aligned} \quad (24)$$

where $K = 255$. Our likelihood ratio selection criteria is to choose the outline O that minimizes $d_L(O, I)$.

5. Classification of an Outline

One of our priorities is to minimize the requirement for human intervention in bone age assessment. The ability to automatically detect whether an outline is

a valid hand is crucial because the later stages of bone extraction and age estimation will fail if the input at the outline stage is incorrect. Error in outlining may be caused by the inaccuracy of the outlining algorithm (none of the approaches described in Sec. 3 are 100% robust against the sources of variation described previously) or by problems with the original image (such as fingers overlapping). In either case, our priority is to avoid incorrect outlines being passed on to the next stage in the process.

The classification task is to predict whether an outline is a valid hand given the outline and the image. We first produced 1000 hand outlines using a mixture of the methods described in Sec. 3. Three volunteers manually labeled the training data as correct or incorrect. Since our priority at this stage is to make sure we do not progress with an incorrectly labeled image, an outline is labeled as correct only if all three human subjects classify it as correct. The training set has 638 positive cases and 362 negative cases.

5.1. Transformation

Generally, image classification requires some form of feature extraction.^{40,41} We extract features in two stages. Firstly, we adopt two fundamentally different representations and, secondly, we derive features from these representations through transformation. As with the ensemble of outlines, the first representation is based on intensity distributions, the second on shape. For intensity features, each segmented image was transformed into two separate intensity distributions, one for outside the outline and one for inside the outline. These two distributions were concatenated to form an instance for each image. Image intensities range from 0–255, but using all of these values may obscure the true differences between the distributions. Hence we created five separate datasets with distributions derived from merging intensity values: Intensity(256) (frequencies for all possible values inside and out, hence 512 features), Intensity(128) (merge every two intensity values), Intensity(64), Intensity(32) and Intensity(16) (merge intensities 0–15,16–32 etc. to give 32 features). The second representation is derived by mapping the outline onto a 1-D series based on the Euclidean distance of each pixel along the contour to the midpoint of the wrist. Each series is smoothed

using a median filter of length 51, z -normalized to remove the scaling effect of age variation and resampled to ensure that each is the same length as the shortest series (2709 attributes). We performed several standard transformations of the 1-D series.

- Principal Component Transforms (PCA). PCA forms a linear transform to an alternative set of orthogonal vectors. We tried two forms of PCA. The first, PCA1, found the components on the whole training set. The second, PCA2, performed the transform on the positive cases only, then used the components to define features for both the positive and negative cases. For both PCA methods we created two data sets. The first contains all the components and the second retained the components that explain 95% of the variation (10 and 14 components, respectively).
- Fast Fourier transforms (FFT). The FFT can be used to capture phase independent information in a series. We used three versions of FFT: FFT (Full) retains all the transformed Fourier terms; FFT (14) kept only the first 14 Fourier terms; and PS further transforms the FFT into the Power Spectrum (by squaring and adding the real and complex Fourier terms).
- Autocorrelation function (ACF). The ACF describes how a series is correlated with itself over a range of different intervals. Thus the k th term of the ACF measures the correlation between each point and the point preceding it by k places.

An alternative approach to classifying raw hand outlines is to derive a number of descriptive features from the data and then use these summary features for classification. Filtering the data to extract bespoke features for classification is a common approach in the literature and is frequently applied to time series and one-dimensional ordered series.⁴² We observe that an outline is often incorrect because it misses a finger, or incorrectly finds a partial bone outline instead of the hand outline. Hence we derive features that relate to the peaks in the series.

There were two main stages in the implementation of the filter. Firstly a simple algorithm was created to detect peaks and troughs in the one-dimensional series that correspond to the tips of the fingers and the webs of the hand. The second stage of the filter uses these landmarks to compute a number of summary measures to represent each hand outline

Algorithm 1: Find Fingers And Webs

Input: One-dimensional Hand Outline \mathbf{Q}
Window Size r

Output: Set of finger locations \mathbf{F}
Set of web locations \mathbf{W}

1. Assume that the initial slope is positive,
 $posSlope = true;$
2. $lastSum = 0;$
3. **FOR** $i = 1$ to $|\mathbf{Q}| - r$
4. $thisSum = \text{sum}(\mathbf{Q}_i \cdots \mathbf{Q}_{i+r})$
5. $difference = thisSum - lastSum$
6. **IF** $posSlope = true$ and $difference < 0$
7. $\mathbf{F.add}(\text{max}(\mathbf{Q}_i \cdots \mathbf{Q}_{i+r}))$
8. $posSlope = false$
9. **ELSEIF** $posSlope = false$ and $difference > 0$
10. $\mathbf{W.add}(\text{min}(\mathbf{Q}_i \cdots \mathbf{Q}_{i+r}))$
11. $posSlope = true$
12. **ENDIF**
13. $lastSum = thisSum$
14. **ENDFOR**
15. **RETURN** \mathbf{F}, \mathbf{W}

in the data set. Algorithm 1 describes the process used to detect the finger tips and webs.

The procedure in Algorithm 1 starts by assuming that the initial slope of the series \mathbf{Q} is positive. A window of size r iteratively moves across \mathbf{Q} from the first position to $|\mathbf{Q}| - r$ by using the loop specified on line 3. For each possible starting location of the window, the total sum of all points within the window is computed and compared to the sum of the previous window (lines 4–5). If the gradient of the slope was previously observed to be positive, the difference between this sum and the last sum must be

positive for this property to remain true; if the difference is negative then the gradient of the line must have changed at some point within the window. This point is identified by finding the local maximum of the window (line 7) and is extracted as a reference to a finger. Once the finger has been extracted, the line direction is updated and the algorithm continues. Conversely, if the line was previously heading in a negative direction, the difference between this window and the previous window must be negative for this to remain true. If line 9 of Algorithm 1 detects that the difference is now positive, the local minimum of the window must correspond to the location of a web of the hand and this location is extracted and added to the set of webs, and the direction of the line is also updated (line 11). The algorithm continues processing the hand outline \mathbf{Q} until it reaches the end of the series, where the sets of fingers and web positions are returned. A graphical illustration of Algorithm 1 is shown in Fig. 13.

Once Algorithm 1 has been performed on all one-dimensional hand outlines, the second phase of the filter can be carried out to transform the finger and web landmarks into a set of descriptive features. We derived 14 features from these positions that fall into four distinct categories: The number of landmarks found (number of fingers, number of webs), relative finger positions to the index finger (thumb to index, middle to index, ring to index, little to index), relative web positions to the thumb/index finger web (index/middle, middle/ring, ring/little), and the ratio of finger height to wrist width (thumb/wrist, index/wrist, middle/wrist, ring/wrist, little/wrist).

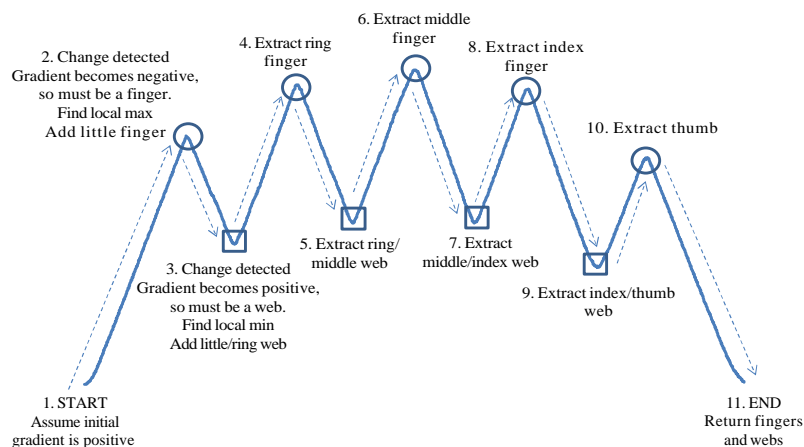


Fig. 13. A graphical illustration of Algorithm 1.

The extraction of these features also allowed for a simple classification rule to be enforced: When given a hand outline to extract features from, if the number of fingers observed by Algorithm 1 is not equal to five and the number of webs extracted is not equal to four, then we can safely classify this hand outline as an incorrect case because all hands in the data must have five fingers and four webs to be a valid hand.

5.2. Classifiers

We conducted our classification experiments on the 15 datasets with 10 different classifiers. We used the WEKA⁴³ implementation of k NN⁴⁴ (with k set through cross validation), C4.5 decision tree, Naive Bayes, Support Vector Machines⁴⁵ with a linear, quadratic and radial basis kernel, Random Forest⁴⁶ with 30 and 100 base classifiers, Rotation Forest⁴⁷ with 30 base classifiers, and a multilayer perceptron. Random Forest and Rotation Forest are tree based ensemble techniques that diversify through random attribute selection (Random Forest) or subspace transformation (Rotation Forest).

6. Results

There are four stages to our experimentation. Firstly, we evaluate classifiers on our training set of outlines and choose a subset of classifiers to use in testing. Secondly, we apply our outlining algorithms to 370 test images. Thirdly, we assess the outline outputs with the classifiers. Finally, we manually assess the outlines and comment on the suitability of the classifiers.

6.1. Classifying outlines

The training set has 638 positive cases and 362 negative cases. The raw (normalized) data has 2709 attributes. All classifiers are assessed through a tenfold cross validation. The mean classification accuracy and standard deviation between folds are shown in Table 1. Generally, building classifiers on transformed data did not improve on the accuracy of those built on the raw data. However, Random Forest with 100 base classifiers achieves the highest overall accuracy of 93.5% using all components of PCA2. A classification accuracy of over 90% is sufficient at this point in the development cycle, hence

we continue to use a Random Forest classifier trained on PCA2 (100%).

6.2. Generating test outlines

The second stage of the experimentation involves forming hand outlines on 370 separate testing images. We used the four methods described in Sec. 3, then run ensembles of the Canny, Otsu and contour algorithms on rescaled images. The AAM is trained on a manually labeled set of 30 images.

6.3. Testing the outlines

We trained a Random Forest classifier with 100 base classifiers using all 1000 of the PCA2 training data, then used this classifier to label the outlines generated by our seven outlining techniques as correct or incorrect. Table 2 shows the percentage of correct outlines for each outlining algorithm, as determined by the Random Forest classifier. Firstly, these results suggest that the AAM technique is the best performing outlining scheme (85% correct) and that Canny is the worst, failing to form a single correct outline. We investigate these results further in Sec. 6.4. Secondly, ensembling with the likelihood ratio method actually makes Otsu and contour worse. When coupled with the fact that the intensity based classifiers performed poorly (see Table 1), this implies that the intensity information is too noisy to use to distinguish outlines. Finally, Table 2 demonstrates that ensembling with DTW improves the performance of both the Otsu and the contour outlining algorithm.

6.4. Manual assessment of outlining algorithms

Table 2 suggests that AAM is the best outliner. In order to explore these results further, we manually labeled the test outlines of AAM and contour ensemble (DTW) as correct or incorrect. This demonstrated that whilst the AAM algorithm usually finds a valid hand shape, this outline is often not in the correct location. Figure 14 gives two examples of this phenomenon.

The fitting procedure used by AAM actually constrains the fit to a hand like shape, so the problem for AAM is finding the location for the constrained outline. Errors occur with AAM when the search algorithm becomes stuck in a local optimum.

Table 1. 10 fold cross validation accuracy (with standard deviation) for 10 classifiers (columns) on fifteen separate data sets (rows).

	k-NN	Naïve Bayes	C4.5 Tree	Linear SVM	Quadratic SVM	Radial SVM	Random Forest (30)	Random Forest (100)	Rotation Forest (30)	Multilayer Perceptron
Raw Data	87.7 (2.1)	83.1 (2.2)	85.8 (2.3)	89.2 (1.7)	89.6 (2.6)	88.2 (1.7)	89.5 (2.3)	89.7 (2.2)	90.8 (2.2)	78.4 (8.2)
FFT (Full)	74.1 (3.0)	70.7 (3.3)	79.4 (2.7)	79.4 (1.6)	80.2 (1.7)	84.1 (2.7)	80.2 (3.4)	80.5 (3.0)	85.6 (1.7)	72.3 (4.2)
FFT (Reduced)	66.5 (2.3)	66.7 (2.5)	66.9 (1.2)	66.8 (2.4)	66.9 (2.4)	64.5 (2.0)	62.0 (4.6)	63.6 (4.4)	67.3 (3.1)	67.2 (2.2)
Power Spectrum	72.8 (3.4)	71.0 (2.9)	76.9 (3.5)	80.0 (2.2)	76.4 (2.2)	71.2 (2.4)	77.4 (3.6)	77.7 (3.4)	84.8 (3.1)	78.3 (3.5)
ACF	80.7 (2.7)	73.4 (2.1)	81.1 (3.7)	85.6 (2.2)	88.5 (2.5)	82.9 (2.9)	86.1 (3.3)	86.4 (3.1)	89.3 (2.5)	79.1 (2.5)
PCA1 (Full)	36.2 (2.0)	80.9 (3.4)	79.7 (3.3)	45.9 (2.5)	46.6 (2.4)	64.2 (2.2)	81.9 (3.6)	81.9 (2.1)	81.7 (3.9)	72.1 (6.1)
PCA1 (95%)	87.3 (3.1)	82.6 (2.5)	85.0 (3.6)	86.3 (3.7)	86.4 (4.4)	63.9 (2.1)	88.6 (3.3)	89.0 (3.7)	89.0 (4.1)	89.5 (4.1)
PCA2 (Full)	80.5 (3.5)	84.8 (1.7)	85.9 (2.1)	86.7 (2.9)	80.3 (2.1)	76.0 (2.1)	93.3 (2.2)	93.5 (2.1)	93.1 (2.3)	81.1 (2.2)
PCA2 (95%)	86.2 (4.5)	84.1 (3.1)	85.1 (3.5)	84.5 (3.9)	85.6 (4.4)	64.3 (2.4)	88.4 (4.6)	87.8 (4.7)	87.0 (4.1)	88.7 (3.9)
Intensity (16)	76.2 (3.6)	67.8 (3.3)	74.6 (3.0)	77.2 (2.6)	73.6 (3.2)	63.8 (2.0)	81.6 (2.2)	82.6 (2.1)	80.1 (3.6)	77.0 (3.4)
Intensity (32)	77.7 (3.7)	66.8 (2.7)	75.0 (4.5)	76.5 (3.0)	75.9 (2.9)	66.5 (3.7)	81.3 (2.1)	82.9 (2.1)	79.4 (2.2)	76.4 (1.5)
Intensity (64)	78.0 (2.5)	67.1 (2.9)	72.7 (4.2)	78.1 (2.7)	75.8 (2.7)	73.1 (2.4)	81.3 (2.5)	83.2 (2.3)	80.5 (2.7)	74.1 (6.7)
Intensity (128)	77.9 (2.5)	66.9 (2.7)	73.7 (3.3)	77.8 (2.1)	75.0 (2.3)	73.9 (2.4)	81.2 (2.6)	82.4 (2.1)	80.5 (2.5)	68.7 (7.7)
Intensity (256)	76.9 (2.5)	68.1 (3.2)	74.9 (3.7)	78.4 (2.7)	77.0 (3.0)	73.6 (2.1)	82.7 (2.6)	82.7 (2.9)	80.4 (2.7)	65.5 (8.9)
Peaks	88.2 (3.0)	75.0 (2.4)	86.0 (3.9)	83.0 (3.0)	86.6 (2.9)	69.0 (2.3)	89.4 (2.9)	89.5 (3.5)	90.0 (2.9)	89.1 (3.3)

In fact, manual inspection revealed that only 187 test images (50.54%) were correctly outlined by AAM. Table 3 shows the confusion matrix of the Random Forest classifier against our manual labeling for AAM outlines. The classifier made 140 false positive classifications.

Since we are primarily concerned with minimizing false positives, this presents a serious problem. We could increase the training set size for AAM and potentially improve performance, but there is a strong likelihood that errors of this nature will still occur. Alternatively we could alter our classification scheme to use a measure derived from the image intensity rather than the outline. However,

Table 2. Percentage of the 370 outlines classified as correct by the Random Forest Classifier (100).

	AAM (%)	Canny (%)	Contour (%)	Otsu (%)
Non-Ensemble	85.68	0.00	25.41	13.78
Ensemble (DTW)	N/A	0.00	77.30	45.41
Ensemble (LLR)	N/A	0.00	1.08	6.48

the intensity based classifiers achieved a maximum 82% accuracy. This indicates that discrimination by intensity distributions is harder than discrimination by shape.

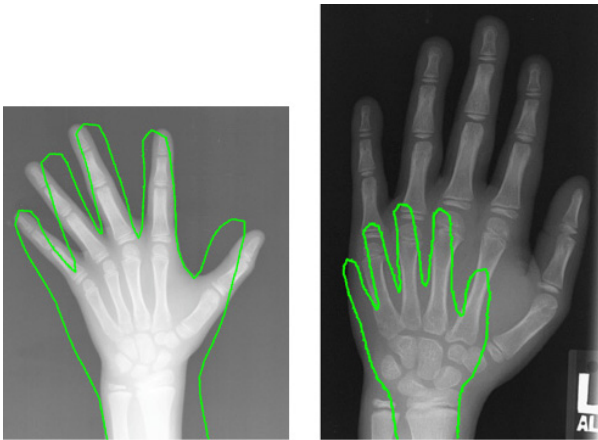


Fig. 14. Two examples of AAM finding an incorrect outline.

Table 3. Confusion matrix for random forest on AAM outlines.

Classified	Actual	
	1	0
1	177	140
0	10	43

The Canny outliner was classified as getting no outlines correct. A visual inspection revealed this was an overly pessimistic scoring, but nevertheless that Canny performed poorly. We believe that this is caused by two factors. Firstly, Canny is an edge detector rather than an outline detector. Whilst a human may classify the Canny output as good, since it broadly looks correct, it does not generally form a continuous outline. Secondly, although the intensity difference between the hand and the background is obvious to the human eye, the actual intensity differentials at the boundaries are not great.

In contrast, a visual inspection of the contour ensemble (DTW) outlines reveals that it correctly

Table 4. Confusion matrices for random forest on contour ensemble (DTW).

Classified	Actual	
	1	0
1	265	21
0	55	29

found 320 outlines (86.46%). Table 4 shows that the Random Forest made only 21 false positive classifications and was cautious about labeling, with over twice the number of false negative as false positives. This is actually desirable, as our primary concern is to stop incorrect outlines proceeding to the bone extraction stage.

Our primary conclusion from these experiments is that the contour ensemble (DTW) is the most appropriate outlining algorithm for hand images, and that Random Forest classifiers using a PCA transformation are the most appropriate way of automatically classifying outlines as correct or not.

7. Conclusions and Future Work

This paper describes a novel ensemble algorithm for outlining radiographs and a classification scheme to automatically detect whether an outline is correct. We found that of the two voting schemes used in the ensemble, DTW outperforms the likelihood ratio test. We have successfully applied this ensemble algorithm to a contouring outliner that can extract correct outlines from over 80% of images. We conclude that AAM is the only other contender in terms of accuracy, but is not suitable for this project because the types of mistake it makes are hard to detect automatically.

While the Random Forest classifier performs adequately, we believe we could construct a better classifier. We will experiment with alternative transformations and classification schemes to try to improve performance. The contour ensemble algorithm performs well, but we aim to improve performance by setting contour levels dynamically.

Pre-processing of hand radiographs is a nontrivial task and is the starting point for automatic diagnoses in many tasks in the field of medical imaging.⁴⁸⁻⁵⁰

A possible area of further work is to investigate other segmentation techniques such as semantic-based techniques. Semantic-based techniques for image segmentation aim to group the pixels of an image into semantically meaningful sets, where the information conveyed by the pixels within a group is similar in some sense. Typically this will involve some form of cluster analysis on the greyscale/color pixel information, the image gradients, and/or the local texture information,⁵¹ or it might involve a

more sophisticated form of clustering and classification using decision trees.⁵² The advantage of these approaches is that they require no *a priori* knowledge and operate on the image data directly. But, there is no guarantee that the image segments align properly with real-world objects. Conversely, model-based methods, e.g. ASMs and AAMs, require training but have the advantage the segments correspond to the actual object of interest.

In the wider context of this project, the next task is to extract individual bones from an outlined hand image. We will concentrate on extracting the phalanges and then attempt to recreate the TW classification scheme by investigating what features are important for bone age assessment.

References

1. J. M. Tanner, R. H. Whitehouse, N. Cameron, W. A. Marshall, M. J. R. Healy and N. H. Goldstein, *Assessment of Skeletal Maturity and Prediction of Adult Height* (TW3 method) (WB Saunders, 2001).
2. H. Thodberg, S. Kreiborg, A. Juul and K. Pedersen, The BoneXpert method for automated determination of skeletal maturity, *IEEE Trans. Med. Imaging* **28** (2009) 52–66.
3. N. Otsu, A threshold selection method from gray-level histograms, *Automatica* **11** (1975) 285–296.
4. J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* **8** (1986) 679–698.
5. T. F. Cootes, G. J. Edwards and C. J. Taylor, Active appearance models, *IEEE Trans. Pattern Anal. Mach. Intell.* **23** (2001) 681–685.
6. R. B. Husar, N. V. Gillani, J. D. Husar, C. C. Paley and P. N. Turcu, Long-range transport of pollutants observed through visibility contour maps, weather maps and trajectory analysis, *Third Symposium on Turbulence, Diffusion and Air Pollution, American Meteorological Society*, Reno, NV, 1976, pp. 344–347.
7. A. Bielecki, M. Korkosz and B. Zielinski, Hand radiographs preprocessing, image representation in the finger regions and joint space width measurements for image interpretation, *Pattern Recognit.* **41** (2008) 3786–3798.
8. T. M. Lehmann, D. Beier, C. Thies and T. Seidl, Segmentation of medical images combining local, regional, global, and hierarchical distances into a bottom-up region merging scheme, *Proc. SPIE* **5747** (2005) 546–555.
9. H. H. Thodberg, Hands-on experience with active appearance models, *Proc. SPIE* **4684** (2002) 495–506.
10. F. Cao, H. K. Huang, E. Pietka and V. Gilsanz, Digital hand atlas and web-based bone age assessment: System design and implementation, *Comput. Med. Imaging Graph.* **24**(5) (2000) 297–307.
11. L. M. Davis, B. J. Theobald, A. Toms and A. Bagnall, On the extraction and classification of hand outlines, *Lect. Notes Comput. Sci.* **6936** (2011) 92–99.
12. C. W. Hsieh, T. C. Liu, J. K. Wang, T. L. Jong and C. M. Tiu, A simplified RUS bone age assessment procedure using grouped-TW method, *Pediatr. Int.* (2011).
13. C. W. Hsieh, T. L. Jong and C. M. Tiu, Bone age estimation based on phalanx information with fuzzy constrain of carpals, *Med. Biol. Eng. Comput.* **45**(3) (2007) 283–295.
14. A. Schmeling and S. Black, An introduction to the history of age estimation in the living, in *Age Estimation in the Living* (John Wiley & Sons, 2010).
15. T. M. Rotch, A study of the development of the bones in childhood by the roentgen method with the view of establishing a developmental index for the grading of and the protection of early life, *Trans. Assoc. Am. Physicians* **24** (1909) 603.
16. R. Cameriere, L. Ferrante, D. Mirtella and M. Cingolani, Carpals and epiphyses of radius and ulna as age indicators, *Int. J. Legal Med.* **120**(3) (2006) 143–146.
17. T. W. Todd, *Atlas of Skeletal Maturation* (Mosby, 1937).
18. W. W. Greulich and S. I. Pyle, *Radiographic Atlas of Skeletal Development of the Hand and Wrist: Based on the Brush Foundation Study of Human Growth and Development* (Stanford University Press, 1950).
19. W. Greulich and S. Pyle, *Radiographic Atlas of Skeletal Development of the Hand and Wrist* (Stanford University Press, 1959).
20. J. M. Tanner, R. J. Whitehouse, W. A. Marshall, M. J. R. Healy and H. Goldstein, *Assessment of Skeletal Maturity and Prediction of Adult Height* (TW2 method), Vol. 16 (Academic Press London, 1975).
21. J. M. Tanner and R. J. Whitehouse, A new system for estimating skeletal maturity from the hand and wrist: With standards derived from a study of 2,600 healthy British children, International Children's Centre, (1962).
22. R. M. Acheson, A method of assessing skeletal maturity from radiographs: A report from the oxford child health survey, *J. Anat.* **88** (1954) 498–508.
23. R. K. Bull, P. D. Edwards, P. M. Kemp, S. Fry and I. A. Hughes, Bone age assessment: A large scale comparison of the Greulich and Pyle, and Tanner and Whitehouse (TW2) methods, *Arch. Dis. Child.* **81**(2) (1999) 172–173.
24. E. Andersen, Skeletal maturation of Danish school children in relation to height, sexual development, and social conditions, *Acta Paediatr. Scand.* **185** (1968) 1–133.
25. R. R. van Rijn, M. H. Lequin, S. G. F. Robben, W. C. J. Hop and C. van Kuijk, Is the Greulich and Pyle

- atlas still valid for Dutch Caucasian children today?, *Pediatr. Radiol.* **31**(10) (2001) 748–752.
26. E. Pietka, S. Pospiech-Kurkowska, A. Gertych and F. Cao, Integration of computer assisted bone age assessment with clinical PACS, *Comput. Med. Imaging Graph.* **27**(2–3) (2003) 217–228.
 27. B. Zielinski, Hand radiograph analysis and joint space location improvement for image interpretation, *Schedae Informaticae* **17**(1) (2009) 45–61.
 28. N. Efford, Knowledge-based segmentation and feature analysis of hand wrist radiographs, *Proc. SPIE* **1905** (1993) 596–608.
 29. S. Mahmoodi, B. Sharif, E. Chester, J. Owen and R. Lee, Skeletal growth estimation using radiographic image processing and analysis, *IEEE Trans. Inf. Technol. Biomed.* **4**(4) (2000) 292–297.
 30. M. Niemeijer, B. van Ginneken, C. Maas, F. Beek and M. Viergever, Assessing the skeletal age from a hand radiograph: automating the Tanner-Whitehouse method, *SPIE Med. Imaging* **5032** (2003) 1197–1205.
 31. S. Adeshina, T. Cootes and J. Adams, Evaluating different structures for predicting skeletal maturity using statistical appearance models, *Proc. Medical Imaging Understanding and Analysis (MIUA)*, (2009) 62–66.
 32. I. Matthews and S. Baker, Active appearance models revisited, *Int. J. Comput. Vis.* **60**(2) (2004) 135–164.
 33. E. Muñoz-Moreno, R. Cárdenes, R. De Luis-García, M. Á. Martín-Fernández and C. Alberola-López, Automatic detection of landmarks for image registration applied to bone age assessment, in *Proc. 5th WSEAS International Conference on Signal Processing, Computational Geometry & Artificial Vision* (World Scientific and Engineering Academy and Society, 2005), pp. 117–122.
 34. Y. Shi and T. Tsui, An FPGA-based smart camera for gesture recognition in HCI applications, in *Proc. 8th Asian conference on Computer Vision*, Vol. 1 (2007) 718–727.
 35. G. Piccioli, E. De Micheli, P. Parodi and M. Campani, Robust road sign detection and recognition from image sequences, *Proceedings of the Intelligent Vehicles Symposium (IEEE)* (1994), pp. 278–283.
 36. R. Scherer, Designing boosting ensemble of relational fuzzy systems, *Int. J. Neural Syst.* **20**(5) (2010) 381–388.
 37. B. Baruque, E. Corchado and H. Yin, The S2-Ensemble fusion algorithm, *Int. J. Neural Syst.* **21**(6) (2011) 505–525.
 38. H. Ding, G. Trajcevski, P. Scheuermann, X. Wang and E. Keogh, Querying and mining of time series data: Experimental comparison of representations and distance measures, *Proceedings of the 34th International Conference on Very Large Databases (VLDB)* (2008), pp. 1542–1552.
 39. V. Gilsanz and O. Ratib, *Hand Bone Age: A Digital Atlas of Skeletal Maturity* (Springer-Verlag, 2005).
 40. W. Zou, Z. Chi and K. C. Lo, Improvement of image classification using wavelet coefficients with structured-based neural network, *Int. J. Neural Syst.* **18**(3) (2008) 195–205.
 41. Md. M. Hyder, Md. M Islam, M. A. H. Akhand and K. Murase, Symmetry axis based object recognition under translation, rotation and scaling, *Int. J. Neural Syst.* **19**(1) (2009) 25–42.
 42. J. Lines, A. Bagnall, P. Caiger-Smith and S. Anderson, Classification of household devices by electricity usage profiles, *Lect. Notes Comput. Sci.* **6936** (2011) 403–412.
 43. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, The WEKA Data Mining Software: An Update, *SIGKDD Explor.* **11**(1) (2009) 10–18.
 44. R. Gil-Pita and X. Yao, Evolving edited k-nearest neighbour classifiers, *Int. J. Neural Syst.* **18**(6) (2008) 459–467.
 45. V. Jumutc, P. Zayakin and A. Borisov, Ranking-based kernels in applied biomedical diagnostics using a support vector machine, *Int. J. Neural Syst.* **21**(6) (2011) 459–473.
 46. L. Breiman, Random forests, *Mach. Learn.* **45**(1) (2001) 5–32.
 47. J. J. Rodriguez, L. I. Kuncheva and C. J. Alonso, Rotation forest: A new classifier ensemble method, *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(10) (2006) 1619–1630.
 48. J. Sharp, J. Gardner and E. Bennett, Computer-based methods for measuring joint space and estimating erosion volume in the finger and wrist joints of patients with rheumatoid arthritis, *Arthritis & Rheum.* **43** (2000) 1378–1386.
 49. R. Tadeusiewicz and M. R. Ogiela, Picture languages in automatic radiological palm interpretation, *Int. J. Appl. Math. Comput. Sci.* **15** (2005) 305–312.
 50. M. R. Ogiela, R. Tadeusiewicz and L. Ogiela, Image languages in intelligent radiological palm diagnostics, *Pattern Recognit.* **39** (2006) 2157–2165.
 51. D. Ilea and P. Whelan, Color image segmentation using a spatial k-means clustering algorithm. International Machine Vision and Image Processing Conference, (2006) 146–153.
 52. J. Shotton, M. Johnson and R. Cipolla, Semantic texton forests for image categorization and segmentation, *Proc. IEEE Conference on Computer Vision and Pattern Recognition* (2008), pp. 1–8.