Automatic spline fitting of planar curvilinear profiles in digital images using the Hough transform

Daniela Schenone
joint work with Costanza Conti and Lucia Romani

Università degli Studi di Milano–Bicocca

daniela.schenone@unimib.it

February 27, 2017

Overview

1. The Hough Transform for detection of
   - Lines
   - Algebraic Curves
2. Standard profile recognition with HT
3. Our profile recognition
4. Numerical tests on medical images: detection of
   - Vertebra
   - Hip bone

Historical overview

The **Hough transform** is a well-established technique used in image analysis and digital image processing to recognize shapes in images.

Historical overview

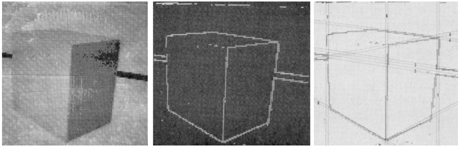
The **Hough transform** is a well-established technique used in image analysis and digital image processing to recognize shapes in images.

1. P. V. C. Hough [1962]: detection of straight lines in images;

Historical overview

The **Hough transform** is a well-established technique used in image analysis and digital image processing to recognize shapes in images.

1. P. V. C. Hough [1962]: detection of straight lines in images;
2. R. O. Duda, P. E. Hart [1972]: better detection of lines, detection of circles and ellipses;



Figures from: R.O. Duda, P.E. Hart, Use of Hough transform to detect lines and curves in pictures [1972].

Historical overview

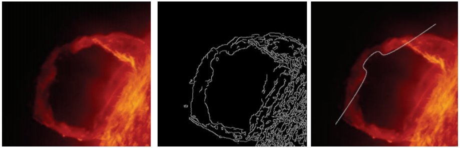
The **Hough transform** is a well-established technique used in image analysis and digital image processing to recognize shapes in images.

1. P. V. C. Hough [1962]: detection of straight lines in images;
2. R. O. Duda, P. E. Hart [1972]: better detection of lines, detection of circles and ellipses;
3. D. Ballard [1981]: detection with template matching;

Historical overview

The **Hough transform** is a well-established technique used in image analysis and digital image processing to recognize shapes in images.

1. P. V. C. Hough [1962]: detection of straight lines in images;
2. R. O. Duda, P. E. Hart [1972]: better detection of lines, detection of circles and ellipses;
3. D. Ballard [1981]: detection with template matching;
4. M. C. Beltrametti, A. M. Massone, M. Piana [2013]: detection of special classes of curves.



Figures from: M.C. Beltrametti, A.M. Massone, M. Piana, Hough transform of special classes of curves [2013].

The goal of our work

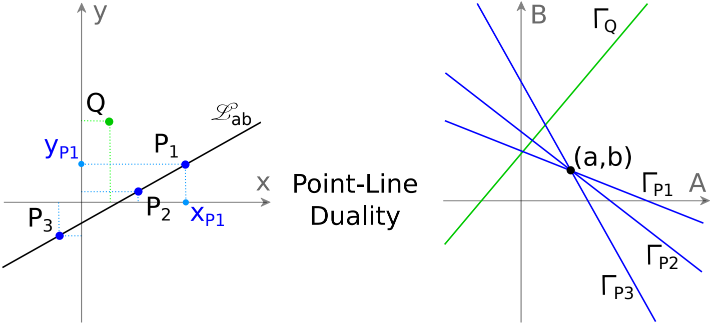
The **goal of our work** is to develop a Hough Transform based algorithm that does not require in input either a family of approximating curves or a predefined look–up table of prototypal shapes.

The outcome of our algorithm is a G^1**–continuous polynomial spline curve**, possibly including some points of C^0–continuity.

Hough transform for family of lines

Definition

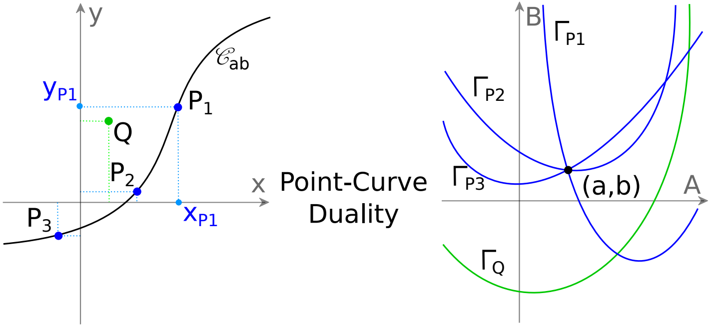
Let $Q = (x_Q, y_Q) \in \mathbb{A}^2_K$ and $\mathcal{L} = \{\ell_{a,b} : Y - aX - b = 0 \mid (a,b) \in K^2\}$, be a family of lines in $\mathbb{A}^2_K(X,Y)$.
The **Hough transform of Q with respect to $\mathcal{L}$** is the line $\Gamma_Q(\mathcal{L})$: $y_Q - Ax_Q - B = 0$, in the parameter plane $\mathbb{A}^2_K(A,B)$.



Hough transform for family of curves

Definition

Let $Q \in \mathbb{A}^2_K$ and let $\mathcal{F}$ be the family of algebraic plane curves
$\mathcal{F} = \{\mathcal{C}_\lambda : F(\lambda; X, Y) = 0 \mid \lambda \in \mathcal{U} \subseteq K^n\}$. (If $n = 2$, $\lambda = (a, b)$).
Then $\Gamma_Q(\mathcal{F}) : F(\Lambda; x_Q, y_Q) = 0$,
is **the Hough transform of the point Q with respect to the family $\mathcal{F}$**.



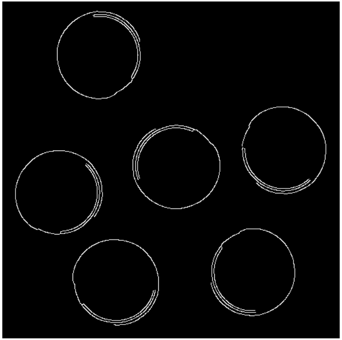
Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

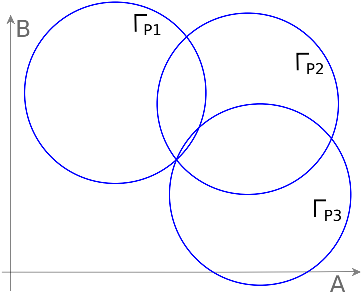
1. Canny edge detection



Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

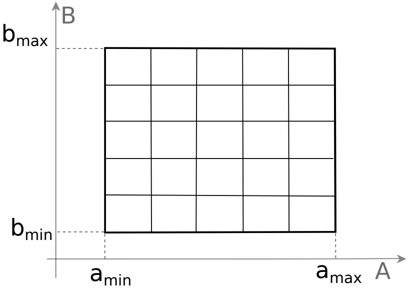
1. Canny edge detection
2. compute Hough transform of edge points



Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

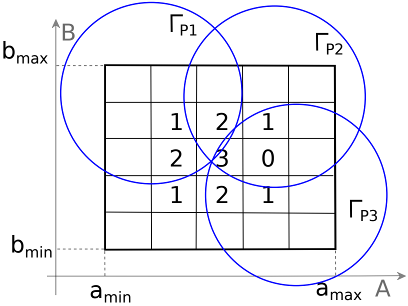
1. Canny edge detection
2. compute Hough transform of edge points
3. discretise parameter space in cells



Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

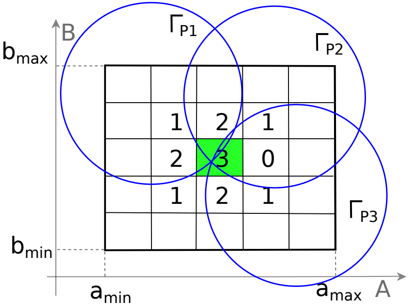
1. Canny edge detection
2. compute Hough transform of edge points
3. discretise parameter space in cells
4. compute accumulation matrix



Standard Hough transform technique algorithm

`input:` image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

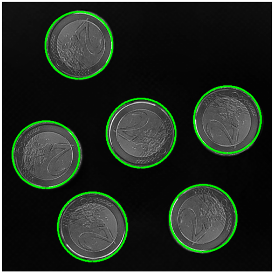
1. Canny edge detection
2. compute Hough transform of edge points
3. discretise parameter space in cells
4. compute accumulation matrix
5. λ^* is the parameter corresponding to the cell with the highest number of HT passing through



Standard Hough transform technique algorithm

`input`: image, family of curves $\mathcal{F} = \{\mathcal{C}_\lambda\}$, with $\lambda = (a, b)$ in this case

1. Canny edge detection
2. compute Hough transform of edge points
3. discretise parameter space in cells
4. compute accumulation matrix
5. λ^* is the parameter corresponding to the cell with the highest number of HT passing through
6. the approximating curve is defined as $\mathcal{C}_{\lambda^*}$



Drawback of existing HT–based methods

All the existing profile detection algorithms based on the Hough transform suffer from the following **drawback**:
they cannot detect unknown shapes since they all require either a family of curves that can approximate the profile or a template for its shape (i.e., a predefined look-up table);

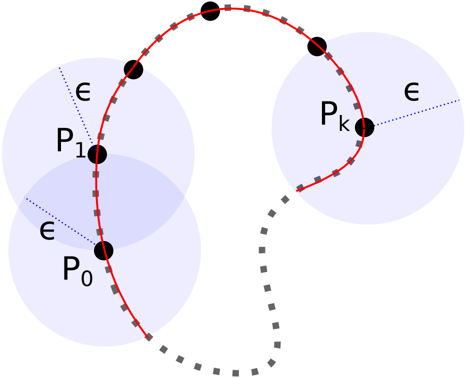
Drawback of existing HT–based methods

All the existing profile detection algorithms based on the Hough transform suffer from the following **drawback**:
they cannot detect unknown shapes since they all require either a family of curves that can approximate the profile or a template for its shape (i.e., a predefined look-up table);

The **advantage** of our new method is that it can match generic shapes in images with a polynomial spline curve without previous knowledge of them.

The idea behind our algorithm

Our idea consists in combining **polynomial splines** with a piecewisely defined **weighted Hough transform**, exploiting only a specific subset of the set of points for each piece of the curve.

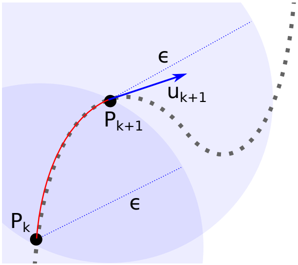


Families of curves

The families of algebraic curves that we use for the piecewise recognition of the profile have the general forms
$\mathcal{F}_{d,X} = \{C^d_\lambda : X = \lambda_d Y^d + \ldots \lambda_1 Y + \lambda_0\}, \quad \lambda_0, \ldots, \lambda_d \in \mathbb{R}$
or
$\mathcal{F}_{d,Y} = \{C^d_\mu : Y = \mu_d X^d + \ldots \mu_1 X + \mu_0\}, \quad \mu_0, \ldots, \mu_d \in \mathbb{R}.$

Families of curves

The families of algebraic curves that we use for the piecewise recognition of the profile have the general forms
$\mathcal{F}_{d,X} = \{C^d_\lambda : X = \lambda_d Y^d + \dots \lambda_1 Y + \lambda_0\}, \quad \lambda_0, \dots, \lambda_d \in \mathbb{R}$
or
$\mathcal{F}_{d,Y} = \{C^d_\mu : Y = \mu_d X^d + \dots \mu_1 X + \mu_0\}, \quad \mu_0, \dots, \mu_d \in \mathbb{R}.$

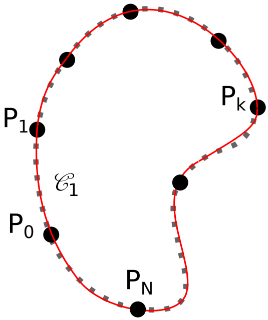


The parameters λ_0 and μ_0 are fixed by the starting point of each curve, λ_1 and μ_1 might be fixed by the tangent versor of the previous curve.

The choice of degree for the families of curves

The choice of the **degree** d depends on the number of constraints on each arc of the curve.
By keeping the number of **free parameters** fixed, let us say 3, and the curve G^1–connected, each piece of curve will have degree $4 \leq d \leq 7$.

degree of $\mathscr{C}_1$ = 3+1

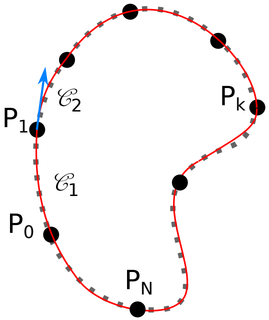


The choice of degree for the families of curves

The choice of the **degree** d depends on the number of constraints on each arc of the curve.
By keeping the number of **free parameters** fixed, let us say 3, and the curve G^1–connected, each piece of curve will have degree $4 \leq d \leq 7$.

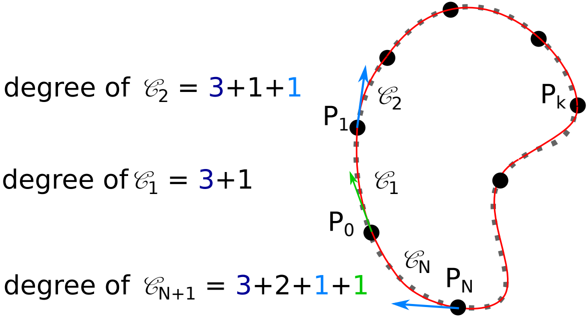
degree of $\mathscr{C}_2$ = 3+1+1

degree of $\mathscr{C}_1$ = 3+1



The choice of degree for the families of curves

The choice of the **degree** d depends on the number of constraints on each arc of the curve.
By keeping the number of **free parameters** fixed, let us say 3, and the curve G^1–connected, each piece of curve will have degree $4 \leq d \leq 7$.



Weighted Hough transform

Instead of using the standard Hough transform technique, we give to **each point a particular weight** in the accumulator matrix.
The weight of an arbitrary point P in its subset is inversely proportional to its squared distance from P_k, starting point of the k^{th} curve arc

$$\omega(P,k) = \frac{1}{d(P,P_k)^2}.$$

In this way, the points closer to the starting point P_k are privileged during each recognition in the algorithm.

In general, when a specific weight in the accumulator matrix is given to each point of the dataset,
the process is called **weighted Hough transform**.

Pre–processing steps

Before starting our algorithm we consider two **pre–processing steps**:

Pre–processing steps

Before starting our algorithm we consider two **pre–processing steps**:

- Application of the Canny edge detection algorithm to the image, to obtain the set of the discontinuity points called $\mathcal{D}$.

Pre–processing steps

Before starting our algorithm we consider two **pre–processing steps**:

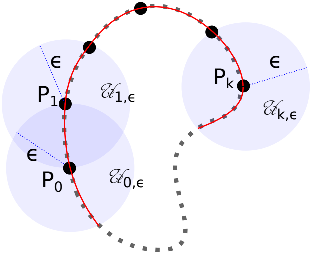
- Application of the Canny edge detection algorithm to the image, to obtain the set of the discontinuity points called $\mathcal{D}$.
- Computation of the local discrete tangent vector for every point of $\mathcal{D}$. If we find points in which there is a sharp change in the orientation of the tangent vector of the profile, the recognized curve will be C^0–continuous (instead of G^1–continuous), as expected.

Free parameter setting

The outcome of our recognition algorithm depends on the choice of the **free parameter** $\varepsilon \in \mathbb{R}_+$.
It is the radius of a disk with center the starting point of the curve arc, and it defines the subset of points, called $\mathcal{U}_{k,\varepsilon}$, to which apply the Hough transform technique.

Free parameter setting

The outcome of our recognition algorithm depends on the choice of the **free parameter** $\varepsilon \in \mathbb{R}_+$.
It is the radius of a disk with center the starting point of the curve arc, and it defines the subset of points, called $\mathcal{U}_{k,\varepsilon}$, to which apply the Hough transform technique.



We may need to set the starting point P_0 in $\mathcal{D}$ for the piecewise curve.
It can be a point with sharp orientation variation of the tangent vector found in the preprocessing step.

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$
    - for each point P_k repeat the following steps

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$
   - for each point P_k repeat the following steps
2. application of Hough transform technique to the subset $\mathcal{U}_{k,\varepsilon}$ with respect to the families $\mathcal{F}_{X,d,P_k}$ and $\mathcal{F}_{Y,d,P_k}$

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$
   - for each point P_k repeat the following steps
2. application of Hough transform technique to the subset $\mathcal{U}_{k,\varepsilon}$ with respect to the families $\mathcal{F}_{X,d,P_k}$ and $\mathcal{F}_{Y,d,P_k}$
3. choose $\mathcal{C}_k$ as the best approximating curve between the two

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$
    - for each point P_k repeat the following steps
2. application of Hough transform technique to the subset $\mathcal{U}_{k,\varepsilon}$ with respect to the families $\mathcal{F}_{X,d,P_k}$ and $\mathcal{F}_{Y,d,P_k}$
3. choose $\mathcal{C}_k$ as the best approximating curve between the two
4. define P_{k+1} as the farthest point arc-length-wise, but still in the disk of radius ε from P_k on the curve $\mathcal{C}_k$

Our algorithm

Finally, this is our **algorithm**.

`input:` $\mathcal{D}, \varepsilon$ and the general families of curves $\mathcal{F}_{X,d}$ and $\mathcal{F}_{Y,d}$

1. choice of P_0 and correspondent local subset of points $\mathcal{U}_{0,\varepsilon}$
    - for each point P_k repeat the following steps
2. application of Hough transform technique to the subset $\mathcal{U}_{k,\varepsilon}$ with respect to the families $\mathcal{F}_{X,d,P_k}$ and $\mathcal{F}_{Y,d,P_k}$
3. choose $\mathcal{C}_k$ as the best approximating curve between the two
4. define P_{k+1} as the farthest point arc-length-wise, but still in the disk of radius ε from P_k on the curve $\mathcal{C}_k$
5. if the new local subset of points, $\mathcal{U}_{k+1,\varepsilon}$, is not empty `go to` 2.

Numerical tests on medical images

We tested our algorithm with real medical images from CT scans.

Numerical tests on medical images

We tested our algorithm with real medical images from CT scans.

In particular, we detected

- G^1–continuous curves to be associated to the external and internal profile of a vertebra,

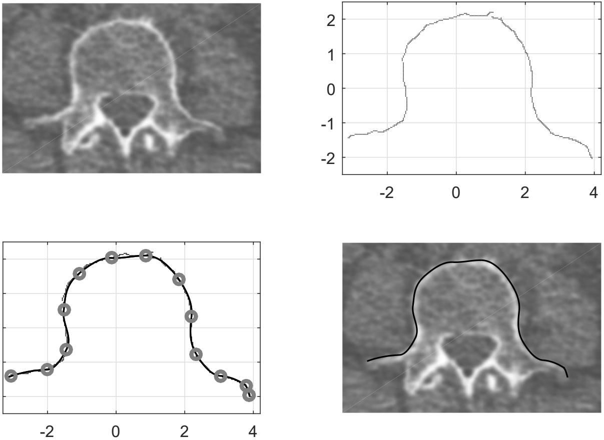
Numerical tests on medical images

We tested our algorithm with real medical images from CT scans.

In particular, we detected

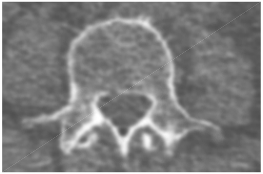
- G^1–continuous curves to be associated to the external and internal profile of a vertebra,
- a G^1–continuous curve, with some C^0 junctions, to be associated to the partial external profile of a hip bone.

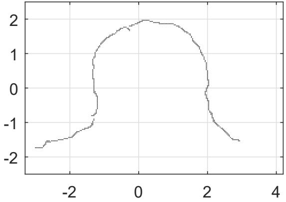
External profile of a vertebra

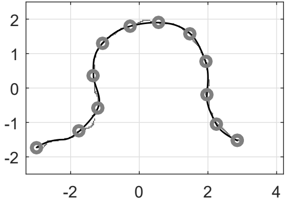


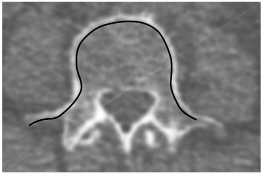
A pixel in the image corresponds to 0.0333 units in the image plane.
MAE $= 0.0250$, RMSE $= 0.0269$.

Internal profile of a vertebra



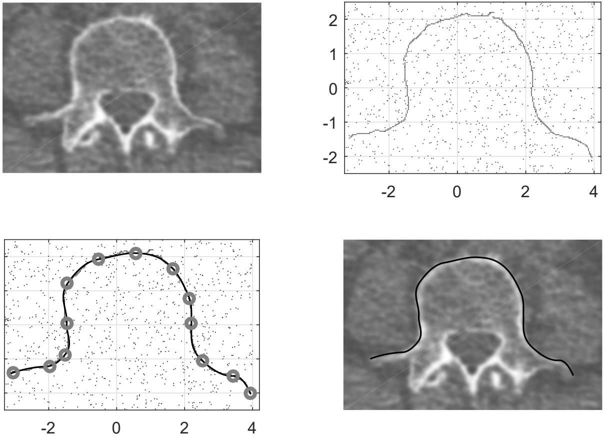






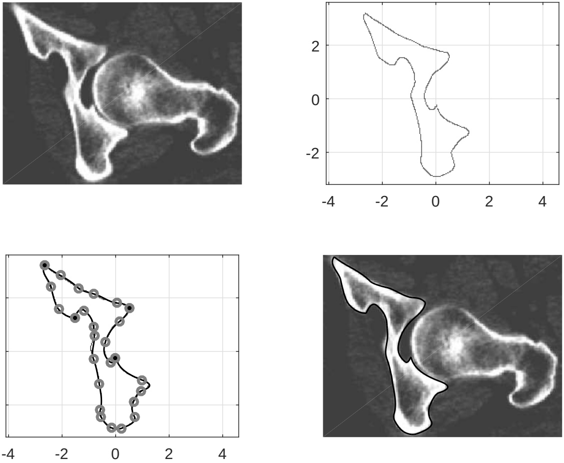
A pixel in the image corresponds to 0.0333 units in the image plane.
MAE 0.0335, RMSE $= 0.0430$.

External profile of a vertebra with noise



A pixel in the image corresponds to 0.0333 units in the image plane. The ratio is 3 noise (uniformly distributed) points for each edge point.

External profile of a hip bone



A pixel in the image corresponds to 0.0333 units in the image plane.
MAE $= 0.0350$, RMSE $= 0.0576$.

Thank you :)