

MEHER KUMAR
MTECH PART TIME
6TH SEMESTER
1906417

CASE STUDY:-

IMAGE PROCESSING SYSTEM FOR LEATHER DESIGNING:-

DIGITAL IMAGE PROCESSING IS THE USE OF DIGITAL COMPUTER TO PROCESS DIGITAL IMAGES THROUGH AN ALGORITHM .EXAMPLE:- ADOBE PHOTOSHOPPING , MATLAB .

STEPS IN IMAGE PROCESSING:-

STEP1- IMAGE ACQUISITION(THE IMAGE IS CAPTURED BY THE SENSOR)

STEP2-IMAGE ENHANCEMENT

STEP3- IMAGE RESTORATION

STEP4- COLOUR IMAGE PROCESSING

STEP5- WAVELETS

STEP6- COMPRESSION

STEP7-MORPHOLOGICAL PROCESSING

STEP8- IMAGE SEGMENTATION

DIGITAL IMAGE PROCESSING CAN INCREASE VISIBILITY OF DETAILS , CAN REDUCE NOISE , CAN ADJUST & OPTIMIZE CONTRAST .

IF INPUT IS IMAGE & DESCRIPTION THEN OUTPUT IS ALSO IMAGE & WITH DESCRIPTION . ALONG WITH DIGITAL IMAGE PROCESSING DONE , WITH COMPUTER VISION & COMPUTER GRAPHICS & ARTIFICIAL INTELLIGENCE .FIRST OF ALL THERE IS PROBLEM DOMAIN .THE IMAGE SENSORS AFTER SENSING THE IMAGE WITH IMAGE PROCESSING HARDWARE & IMAGE PROCESSING SOFTWARE APPLIED ON , COMPUTER THE HARD COPY DEVICE & ALSO A MASS STORAGE DEVICE DISPLAY IMAGE .IMAGE CAPTURED BY CAMERA & SENT TO DIGITAL IMAGE PROCESSING SYSTEM THAT GIVES ITS OUTPUT AS AN PROCESSED IMAGE .AFTER ACQUISITION OF IMAGE , PROCESSING DONE & A COMPARISON IS MADE BASED ON KNOWLEDGE & ASSOCIATIVE STORAGE RESULTING AS AN OUTPUT IMAGE .

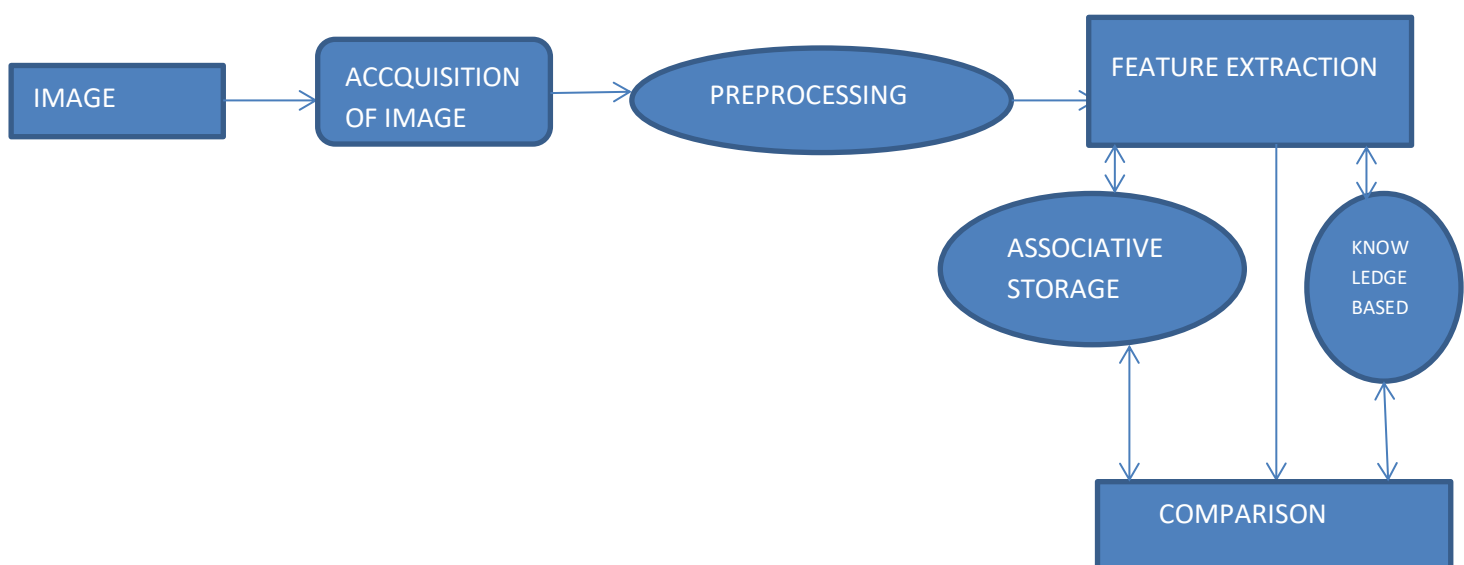


IMAGE PROCESSING WHICH IS A LOW LEVEL PROCESS REDUCE NOISE , DOES CONTRAST , ENHANCEMENT ,
IMAGE SHARPENING . THEN IMAGE ANALYSIS DONE WHICH IS A MID LEVEL PROCESS . IT DOES
SEGMENTATION & CLASSIFICATION . THEN THERE IS VISION DERIVED WHICH IS A HIGH LEVEL PROCESS . IT IS
MAKING SENSE OF AN ENSEMBLE OF RECOGNISED OBJECTS .

ARTIFICIAL INTELLIGENCE FOR IMAGE PROCESSING :-

1. FIRST OF ALL IMAGE ACQUISITION

2. IMAGE ENHANCEMENT

3. IMAGE RESTORATION

4. MORPHOLOGICAL PROCESSING

5. IMAGE RECOGNITION

6. REPRESENTATION & DESCRIPTION

ALL THERE IS PROBLEM DOMAIN THEN THERE IS COLOUR IMAGE PROCESSING & THEN IMAGE COMPRESSION &
DECOMPRESSION.

SOME TECHNIQUES THAT ARE USED FOR DIGITAL IMAGE PROCESSING ARE:-

1. ANISOTROPIC DIFFUSION

2. HIDDEN MARKOV MODELS

3. IMAGE EDITING

4. IMAGE RESTORATION

5. INDEPENDENT COMPONENT ANALYSIS

6. LINEAR FILTERING

7. NEURAL NETWORKS

8. PARTIAL DIFFERENTIAL EQUATIONS

1. ANISTROPIC DIFFUSION:-

IN IMAGE PROCESSING & COMPUTER VISION , ANISOTROPIN DIFFUSION ALSO CALLED PERONA
–MALIK DIFFUSION , IS A TECHNIQUE AIMING AT REDUCING IMAGE NOISE WITHOUT REMOVING
SIGNIFICANT PARTS OF THE IMAGE CONTENT , TYPICALLY EDGES , LINES OR OTHER DETAILS
THAT ARE IMPORTANT FOR THE INTERPRETATION OF THE IMAGE.

2. HIDDEN MARKOV MODELS:-

IT IS A STATISTICAL MODEL (HMM)THAT CAN BE USED TO DESCRIBE THE EVOLUTION OF THE
OBSERVABLE EVENTS THAT DEPEND ON INTERNAL FACTORS , WHICH ARE NOT DIRECTLY
OBSERVABLE.

3. IMAGE EDITING:-

IT REFERS TO MODIFYING OR IMPROVING DIGITAL OR TRADITIONAL PHOTOGRAPHIC IMAGES
USING DIFFERENT TECHNIQUES , TOOLS OR SOFTWARE . IMAGE EDITING IS DONE TO CREATE
THE BEST POSSIBLE LOOK FOR THE IMAGES & ALSO TO IMPROVE THE OVERALL QUALITY OF THE IMAGE
ACCORDING TO DIFFERENT PARAMETERS .

4. IMAGE RESTORATION:-

IT IS THE OPERATION OF TAKING A CORRUPT/NOISY IMAGE & ESTIMATING THE CLEAN ,ORIGINAL IMAGE. CORRUPTION MAY COME IN MANY FORMS SUCH AS MOTION BLUR , NOISE & CAMERA MIS-FOCUS .

5. WAVELETS:-

THE WAVELET APPLICATION INCLUDE NUMERICAL ANALYSIS ,SIGNAL ANALYSIS ,CONTROL APPLICATION ,AND THE ANALYSIS AND ADJUSTMENT OF AUDIO SIGNALS .

6. LINEAR FILTERING-

IT IS THE MOST POWERFUL IMAGE ENHANCEMENT METHODS. IT IS A PROCESS IN WHICH PART OF THE SIGNAL FREQUENCY SPECTRUM IS MODIFIED BY THE TRANSFER FUNCTION OF THE FILTER .

7. NEURAL NETWORKS:-

A NEURAL NETWORK IS A SERIES OF ALGORITHMS THAT ENDEAVORS TO RECOGNIZE UNDERLYING RELATIONSHIPS IN A SET OF DATA THROUGH A PROCESS THAT MIMICS THE WAY THE HUMAN BRAIN OPERATES . IN THIS SENSE NEURAL NETWORKS REFER TO SYSTEM OF NEURONS , EITHER ORGANIC OR ARTIFICIAL IN NATURE .

8. PARTIAL DIFFERENTIAL EQUATIONS:-

PDE SURFACES ARE USED IN GEOMETRIC MODELLING AND COMPUTER GRAPHICS FOR CREATING SMOOTH SURFACES CONFORMING TO A GIVEN BOUNDARY CONFIGURATION .

PDE SURFACES USE PARTIAL DIFFERENTIAL EQUATIONS TO GENERATE A SURFACE WHICH USUALLY SATISFY A MATHMATICAL BOUNDARY VALUE PROBLEM.

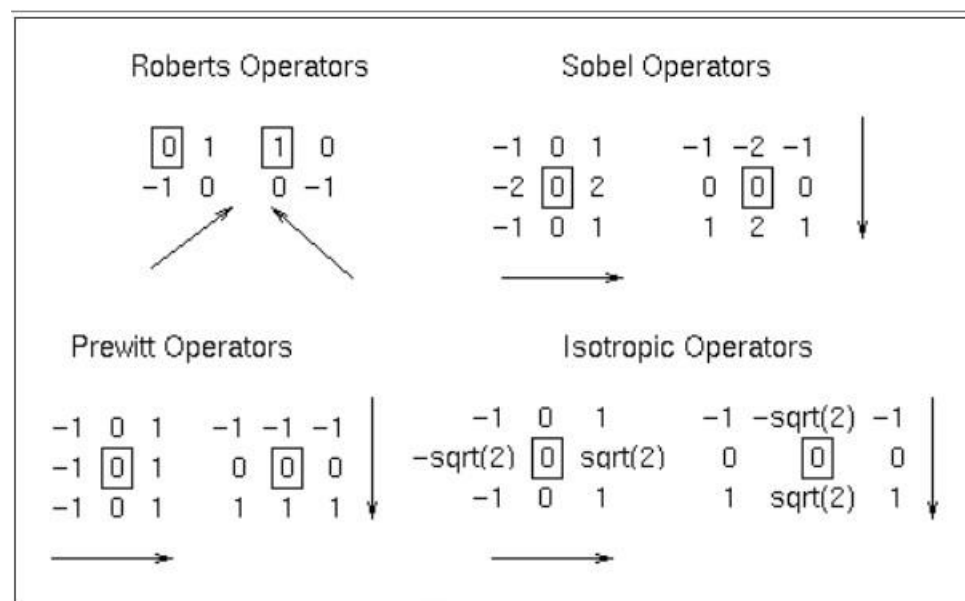
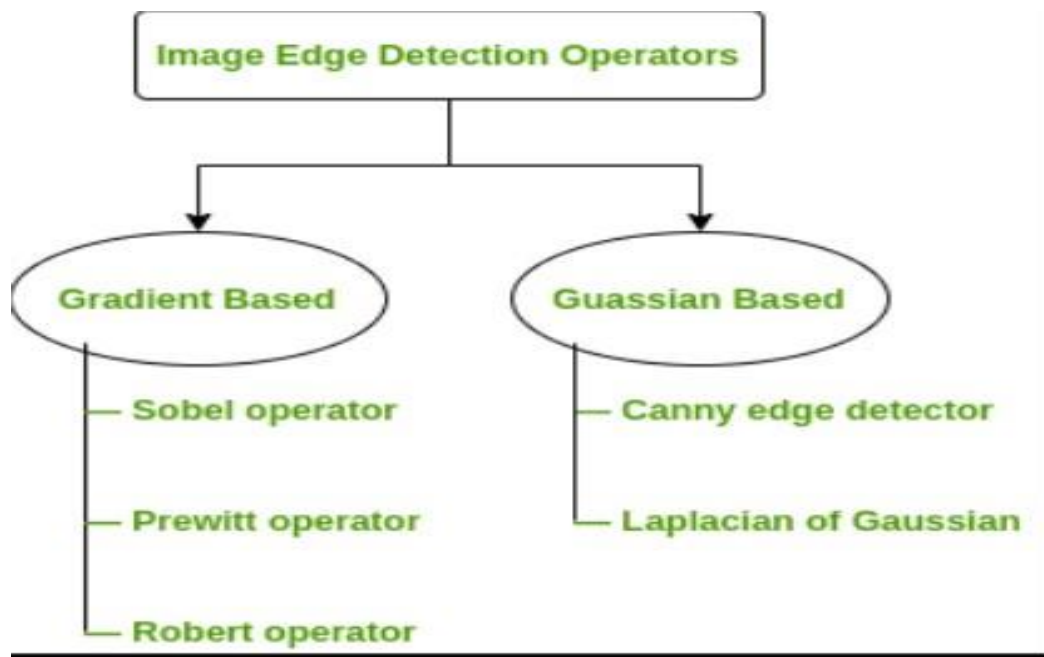
OTHER METHODS:-

KERNEL IMAGE PROCESSING:-

IN IMAGE PROCESSING , A KERNEL CONVOLUTION MATRIX OR MASK IS A SMALL MATRIX . IT IS USED FOR BLURRING , SHARPENING ,EDGE DETECTION AND MORE .THIS IS ACCOMPLISHED BY DOING A C ONVOLUTION BETWEEN A KERNEL & AN IMAGE . THEY ARE ALSO USED IN MACHINE LEARNING FOR 'FEATURE EXTRACTION', A TECHNIQUE FOR DETERMINING THE MOST IMPORTANT PORTIONS OF AN IMAGE .

KERNEL FILTER PROVIDE LOW & HIGH PASS FILTERING (SMOOTHING & SHARPENING RESPECTIVELY) USING A KERNEL . THE FILTER REMOVES ANY PIXELS THAT ARE DARKER THAN A CERTAIN FRACTION OF THE DARKEST NEIGHBORING PIXEL . THE FRACTION IS DETERMINED BY ENTERING A THRESHOLD LEVEL IN PERCENT .

A KERNEL REFERS TO A 2D ARRAY OF WEIGHTS . THE TERM " FILTER " IS FOR 3D STRUCTURES OF MULTIPLE KERNELS STACKED TOGETHER . FOR A 2D FILTER , FILTER IS SAME AS KERNEL . BUT FOR A 3D FILTER & MOST CONVOLUTIONS IN DEEP LEARNING , A FILTER IS COLLECTION OF KERNELS .



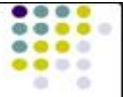
Gradient Operators

SOBEL OPERATOR:-

SOMETIMES CALLED THE SOBEL FIELDMAN OPERATOR OR SOBEL FILTER IS USED IN IMAGE PROCESSING & COMPUTER VISION PARTICULARLY WITHIN EDGE DETECTION ALGORITHM WHERE IT CREATES AN IMAGE EMPHASISING EDGES . IT IS NAMED AFTER IRWIN SOBEL & GARY FELDMAN . AT EACH POINT IN IMAGE, THE RESULT OF THE SOBEL FELDMAN OPERATOR IS EITHER THE CORRESPONDING GRADIENT VECTOR OR THE NORM OF THIS VECTOR . THE SOBEL FELDMAN OPERATOR IS BASED ON CONVOLVING THE IMAGE WITH A SMALL SEPARABLE & INTEGER VALUED FILTER IN THE HORIZONTAL & VERTICAL DIRECTION & IS THEREFORE RELATIVELY INEXPENSIVE IN TERMS OF COMPUTATION . ON THE OTHER HAND , THE GRADIENT APPROXIMATION THAT IT PRODUCES IS RELATIVELY CRUDE , IN PARTICULAR FOR HIGH FREQUENCY VARIATION IN THE IMAGE .



Sobel Operator



- Similar to Prewitt, but averaging kernel is higher in middle

$$H_x^S = \frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * [0.5 \ 0 \ -0.5] = \frac{1}{8} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

$$H_y^S = \frac{1}{4} [1 \ 2 \ 1] * \begin{bmatrix} 0.5 \\ 0 \\ -0.5 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Average in x direction

Derivative in y direction

Note: Filter kernel is flipped in convolution

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Sobel filter for vertical edge detection

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Sobel filter for horizontal edge detection

Gradient Methods – Sobel Operator

- Detection of horizontal & vertical edges
- Convolution Mask
- Masks of size 3×3 matrices is obtain by following eqⁿ
- $G_x = (Z_3 + 2Z_6 + 7Z_9) - (Z_1 + 2Z_4 + Z_7)$ & $G_y = (Z_1 + 2Z_2 + Z_3) - (Z_7 + 2Z_8 + Z_9)$

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

- Smoothing the image by some amount , less susceptible to noise. But it produces thicker edges. So edge localization is poor

GAUSSIAN BLUR:-

IN IMAGE PROCESSING , A GAUSSIAN BLUR , ALSO KNOWN AS GAUSSIAN SMOOTHING , IS THE RESULT OF BLURRING AN IMAGE BY A GAUSSIAN FUNCTION (NAMED AFTER MATHEMATICIAN & SCIENTIST CARL FRIEDRICH GAUSS. IT IS WIDELY USED EFFECT IN GRAPHICS SOFTWARE , TYPICALLY TO REDUCE IMAGE NOISE & REDUCE DETAIL . THE VISUAL EFFECT OF THIS BLURRING TECHNIQUE IS A SMOOTH BLUR RESEMBLING THAT OF VIEWING . THE IMAGE THROUGH A TRANSLUSCENT SCREEN , DISTINCTLY DIFFERENT FROM THE BOKEH EFFECT PRODUCED BY AN OUT OF FOCUS LENS OR THE SHADOW OF AN OBJECT UNDER USUAL ILLUMINATION . GAUSSIAN SMOOTHING IS ALSO USED AS A PRE-PROCESSING STAGE IN COMPUTER VISION ALGORITHM IN ORDER TO ENHANCE IMAGE STRUCTURES AT DIFFERENT SCALES.

IMAGE PROCESSING SYSTEM FOR LEATHER DESIGNING

USING MATLAB:-

TEXTURE SEGMENTATION USING GABOR FILTERS:-

TEXTURE SEGMENTATION IS DONE TO IDENTIFY REGIONS BASED ON THEIR TEXTURE. THE GOAL IS TO SEGMENT THE DOG FROM THE BATHROOM FLOOR. THE SEGMENTATION IS VISUALLY OBVIOUS BECAUSE OF THE DIFFERENCE IN TEXTURE BETWEEN THE REGULAR, PERIODIC PATTERN OF THE BATHROOM FLOOR, AND THE REGULAR, SMOOTH TEXTURE OF THE DOG'S FUR.

FROM EXPERIMENTATION, IT IS KNOWN THAT GABOR FILTERS ARE A REASONABLE MODEL OF SIMPLE CELLS IN THE MAMMALIAN VISION SYSTEM. BECAUSE OF THIS, GABOR FILTERS ARE THOUGHT TO BE A GOOD MODEL OF HOW HUMANS DISTINGUISH TEXTURE, AND ARE THEREFORE A USEFUL MODEL TO USE WHEN DESIGNING ALGORITHMS TO RECOGNIZE TEXTURE.

READ AND DISPLAY INPUT IMAGE :-

Read and display the input image. It shrinks the image to make the example run more quickly.

```
A = imread('kobi.png');
```

```
A = imresize(A,0.25);
```

```
Agray = rgb2gray(A);
```

Figure

```
imshow(A)
```



DESIGN ARRAY OF GABOR FILTERS:-

DESIGN AN ARRAY OF GABOR FILTERS WHICH ARE TUNED TO DIFFERENT FREQUENCIES AND ORIENTATIONS. THE SET OF FREQUENCIES AND ORIENTATIONS IS DESIGNED TO LOCALIZE DIFFERENT, ROUGHLY ORTHOGONAL, SUBSETS OF FREQUENCY AND ORIENTATION INFORMATION IN THE INPUT IMAGE. REGULARLY SAMPLE ORIENTATIONS BETWEEN $[0,150]$ DEGREES IN STEPS OF 30 DEGREES. SAMPLE WAVELENGTH IN INCREASING POWERS OF TWO STARTING FROM $4/\sqrt{2}$ UP TO THE HYPOTENUSE LENGTH OF THE INPUT IMAGE.

```
imageSize = size(A);
```

```
numRows = imageSize(1);
```

```
numCols = imageSize(2);
```

```
wavelengthMin = 4/sqrt(2);
```

```
wavelengthMax = hypot(numRows,numCols);
```

```
n = floor(log2(wavelengthMax/wavelengthMin));
```

```
wavelength = 2.^(0:(n-2)) * wavelengthMin;
```

deltaTheta = 45;

orientation = 0:deltaTheta:(180-deltaTheta);

g = gabor(wavelength,orientation);

EXTRACT GABOR MAGNITUDE FEATURES FROM SOURCE IMAGE. WHEN WORKING WITH GABOR FILTERS, IT IS COMMON TO WORK WITH THE MAGNITUDE RESPONSE OF EACH FILTER. GABOR MAGNITUDE RESPONSE IS ALSO SOMETIMES REFERRED TO AS "GABOR ENERGY". EACH MXN GABOR MAGNITUDE OUTPUT IMAGE IN GABORMAG(:, :, IND) IS THE OUTPUT OF THE CORRESPONDING GABOR FILTER G(IND).

gabormag = imgaborfilt(Agray,g);

POST-PROCESS THE GABOR MAGNITUDE IMAGES INTO GABOR FEATURES :-

TO USE GABOR MAGNITUDE RESPONSES AS FEATURES FOR USE IN CLASSIFICATION, SOME POST-PROCESSING IS REQUIRED. THIS POST PROCESSING INCLUDES GAUSSIAN SMOOTHING, ADDING ADDITIONAL SPATIAL INFORMATION TO THE FEATURE SET, RESHAPING OUR FEATURE SET TO THE FORM EXPECTED BY THE PCA AND KMEANS FUNCTIONS, AND NORMALIZING THE FEATURE INFORMATION TO A COMMON VARIANCE AND MEAN.

EACH GABOR MAGNITUDE IMAGE CONTAINS SOME LOCAL VARIATIONS, EVEN WITHIN WELL SEGMENTED REGIONS OF CONSTANT TEXTURE. THESE LOCAL VARIATIONS WILL THROW OFF THE SEGMENTATION. WE CAN COMPENSATE FOR THESE VARIATIONS USING SIMPLE GAUSSIAN LOW-PASS FILTERING TO SMOOTH THE GABOR MAGNITUDE INFORMATION. WE CHOOSE A SIGMA THAT IS MATCHED TO THE GABOR FILTER THAT EXTRACTED EACH FEATURE. WE INTRODUCE A SMOOTHING TERM K THAT CONTROLS HOW MUCH SMOOTHING IS APPLIED TO THE GABOR MAGNITUDE RESPONSES.

for i = 1:length(g)

sigma = 0.5*g(i).Wavelength;

K = 3;

gabormag(:, :, i) = imgaussfilt(gabormag(:, :, i), K*sigma);

end

WHEN CONSTRUCTING GABOR FEATURE SETS FOR CLASSIFICATION , IT IS USEFUL TO ADD A MAP OF SPATIAL LOCATION INFORMATION IN BOTH X AND Y. THIS ADDITIONAL INFORMATION ALLOWS THE CLASSIFIER TO PREFER GROUPINGS WHICH ARE CLOSE TOGETHER SPATIALLY .

```
X = 1:numCols;
```

```
Y = 1:numRows;
```

```
[X,Y] = meshgrid(X,Y);
```

```
featureSet = cat(3,gabormag,X);
```

```
featureSet = cat(3,featureSet,Y);
```

RESHAPE DATA INTO A MATRIX X OF THE FORM EXPECTED BY THE KMEANS FUNCTION . EACH PIXEL IN THE IMAGE GRID IS A SEPARATE DATAPOINT, AND EACH PLANE IN THE VARIABLE FEATURESET IS A SEPARATE FEATURE. IN THIS EXAMPLE, THERE IS A SEPARATE FEATURE FOR EACH FILTER IN THE GABOR FILTER BANK, PLUS TWO ADDITIONAL FEATURES FROM THE SPATIAL INFORMATION THAT WAS ADDED IN THE PREVIOUS STEP. IN TOTAL, THERE ARE 24 GABOR FEATURES AND 2 SPATIAL FEATURES FOR EACH PIXEL IN THE INPUT IMAGE.

```
numPoints = numRows*numCols;
```

```
X = reshape(featureSet,numRows*numCols,[]);
```

NORMALIZE THE FEATURES TO BE ZERO MEAN, UNIT VARIANCE.

```
X = bsxfun(@minus, X, mean(X));
```

```
X = bsxfun(@divide,X,std(X));
```

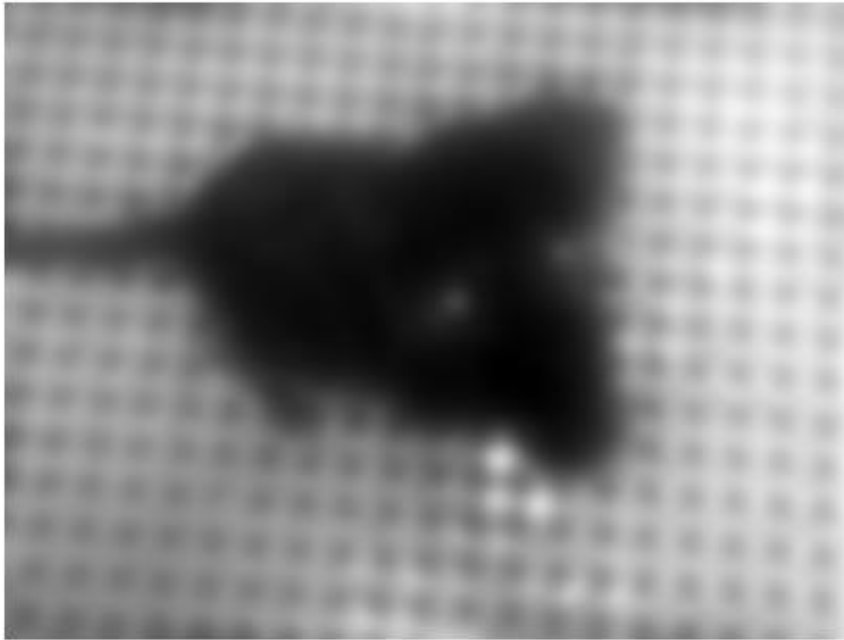
VISUALIZE THE FEATURE SET. TO GET A SENSE OF WHAT THE GABOR MAGNITUDE FEATURES LOOK LIKE, PRINCIPAL COMPONENT ANALYSIS CAN BE USED TO MOVE FROM A 26-D REPRESENTATION OF EACH PIXEL IN THE INPUT IMAGE INTO A 1-D INTENSITY VALUE FOR EACH PIXEL.

```
coeff = pca(X);
```

```
feature2DImage = reshape(X*coeff(:,1),numRows,numCols);
```

```
figure
```

```
imshow(feature2DImage,[])
```



IT IS APPARENT IN THIS VISUALIZATION THAT THERE IS SUFFICIENT VARIANCE IN THE GABOR FEATURE INFORMATION TO OBTAIN A GOOD SEGMENTATION FOR THIS IMAGE . THE DOG IS VERY DARK COMPARED TO THE FLOOR BECAUSE OF THE TEXTURE DIFFERENCES BETWEEN THE DOG AND THE FLOOR .

CLASSIFY GABOR TEXTURE FEATURES USING KMEANS:-

REPEAT K-MEANS CLUSTERING FIVE TIMES TO AVOID LOCAL MINIMA WHEN SEARCHING FOR MEANS THAT MINIMIZE OBJECTIVE FUNCTION. THE ONLY PRIOR INFORMATION ASSUMED IN THIS EXAMPLE IS HOW MANY DISTINCT REGIONS OF TEXTURE ARE PRESENT IN THE IMAGE BEING SEGMENTED. THERE ARE TWO DISTINCT REGIONS IN THIS CASE. THIS PART OF THE EXAMPLE REQUIRES THE STATISTICS AND MACHINE LEARNING TOOLBOX™.

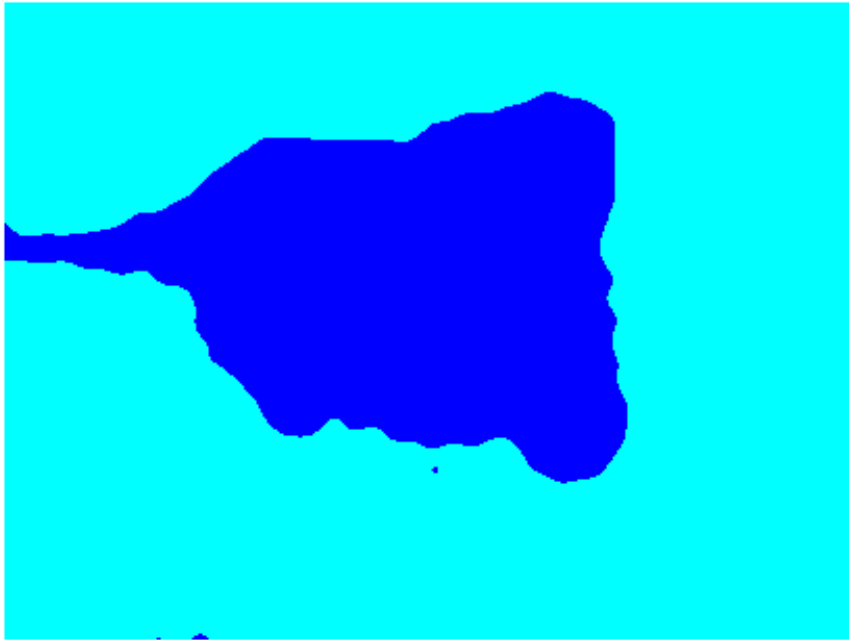
```
L = kmeans(X,2,'Replicates',5);
```

VISUALIZE SEGMENTATION USING LABEL2RGB.

```
L = reshape(L,[numRows numCols]);
```

Figure

```
imshow(label2rgb(L))
```



VISUALIZE THE SEGMENTED IMAGE USING `IMSHOWPAIR`. EXAMINE THE FOREGROUND AND BACKGROUND IMAGES THAT RESULT FROM THE MASK `BW` THAT IS ASSOCIATED WITH THE LABEL MATRIX `L`.

```
Aseg1 = zeros(size(A),'like',A);
```

```
Aseg2 = zeros(size(A),'like',A);
```

```
BW = L == 2;
```

```
BW = repmat(BW,[1 1 3]);
```

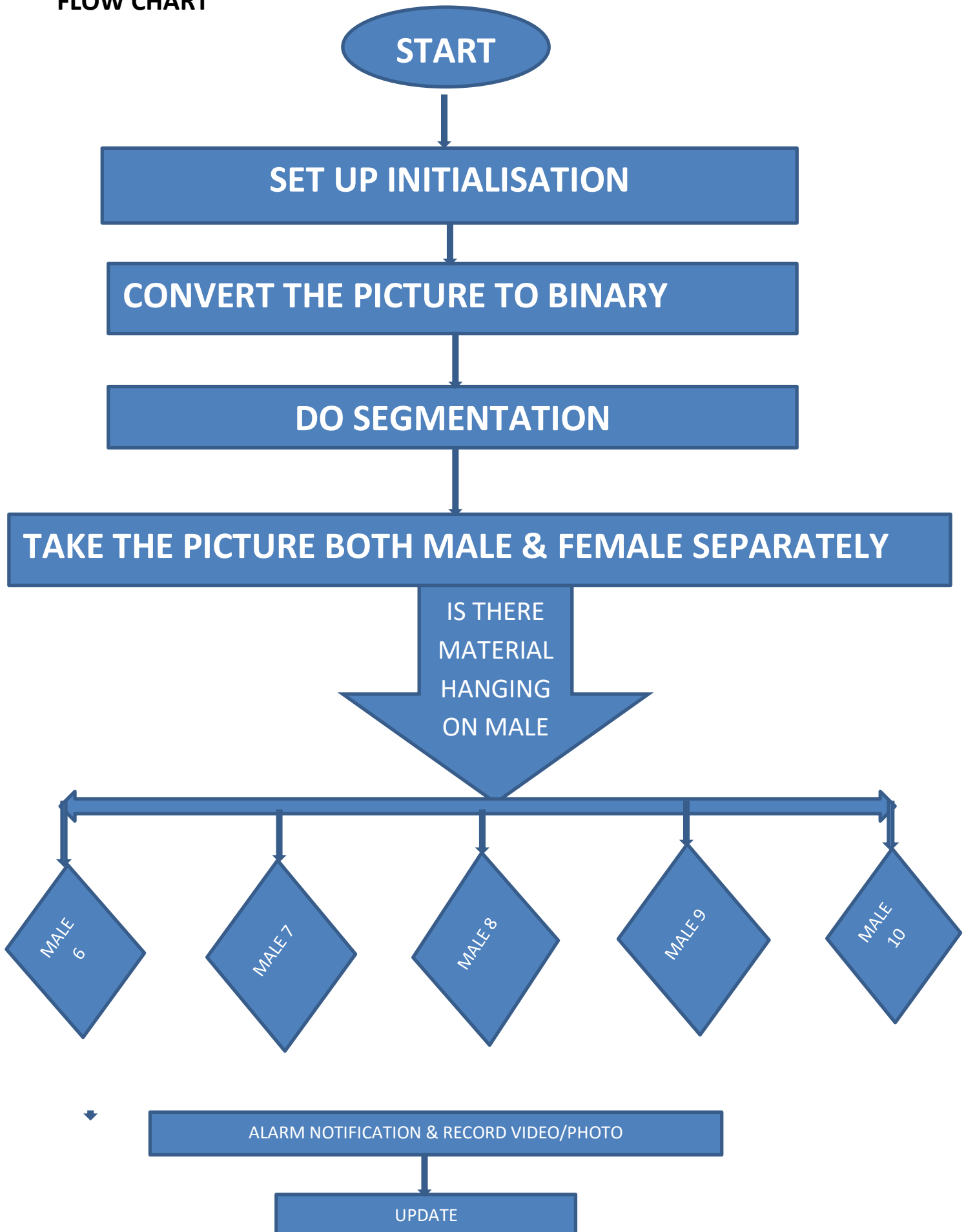
```
Aseg1(BW) = A(BW);
```

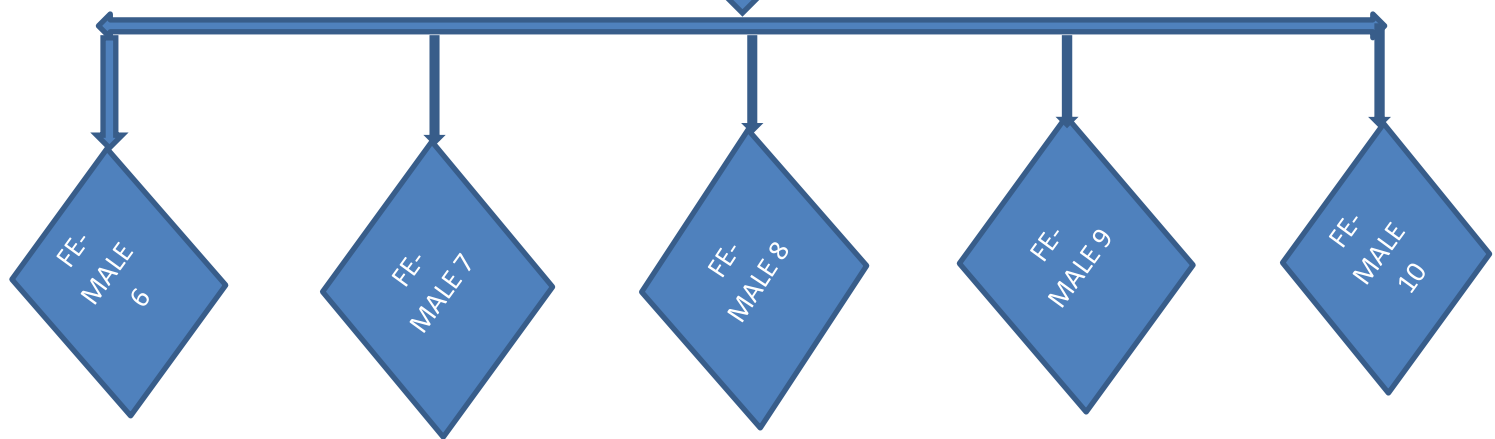
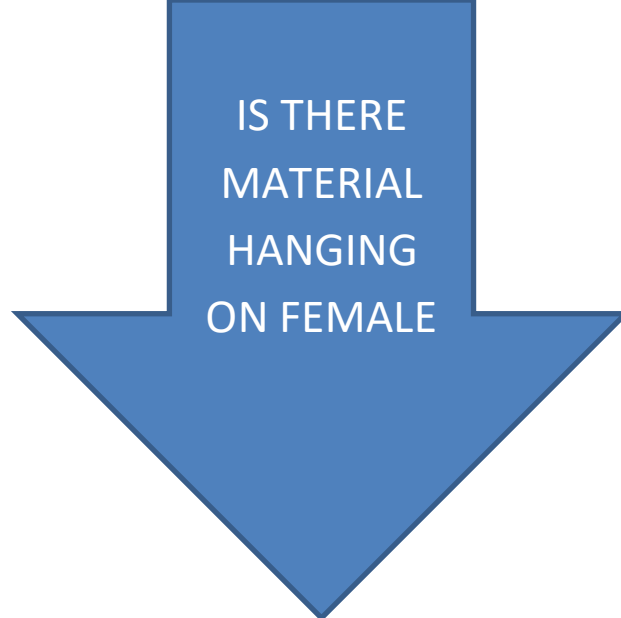
```
Aseg2(~BW) = A(~BW);
```

Figure

```
imshowpair(Aseg1,Aseg2,'montage');
```


FLOW CHART





PROGRAM:-

A = imread('picture_input.bmp');

size (A)

A = imresize(A,0.75);

A=A(1:500,1:500, 1:3);

Agray = rgb2gray(A);

figure(1)

imshow(A)

imageSize = size(A);

figure (2)

imshow(Agray)

numRows = imageSize(1);

numCols = imageSize(2);

figure

wavelengthMin = 4/sqrt(2);

wavelengthMax = hypot(numRows,numCols);

n = floor(log2(wavelengthMax/wavelengthMin));

```

wavelength = 2.^(0:(n-2)) * wavelengthMin;

deltaTheta = 45;

orientation = 0:deltaTheta:(180-deltaTheta);

g = gabor(wavelength,orientation);

gabormag = imgaborfilt(Agray,g);

for i = 1:length(g)

    sigma = 0.5*g(i).Wavelength;

    K = 3;

    gabormag(:, :, i) = imgaussfilt(gabormag(:, :, i), K*sigma);

end

X = 1:numCols;

Y = 1:numRows;

[X,Y] = meshgrid(X,Y);

featureSet = cat(3,gabormag,X);

featureSet = cat(3,featureSet,Y);

numPoints = numRows*numCols;

X = reshape(featureSet,numRows*numCols,[]);

X = bsxfun(@minus, X, mean(X));

X = bsxfun(@rdivide,X,std(X));

```

```
coeff = pca(X);
```

```
feature2DImage = reshape(X*coeff(:,1),numRows,numCols);
```

```
figure (3)
```

```
L =imshow(feature2DImage,[])
```

```
kmeans(X,2,'Replicates',5);
```

```
% L = reshape(L,[numRows numCols]);
```

```
figure (4)
```

```
%imshow(label2rgb(L))
```

```
Aseg1 = zeros(size(A),'like',A);
```

```
Aseg2 = zeros(size(A),'like',A);
```

```
BW = L == 2;
```

```
BW = repmat(BW,[1 1 3]);
```

```
Aseg1(BW) = A(BW);
```

```
Aseg2(~BW) = A(~BW);
```

```
figure(5)
```

```
imshowpair(Aseg1,Aseg2,'montage');
```

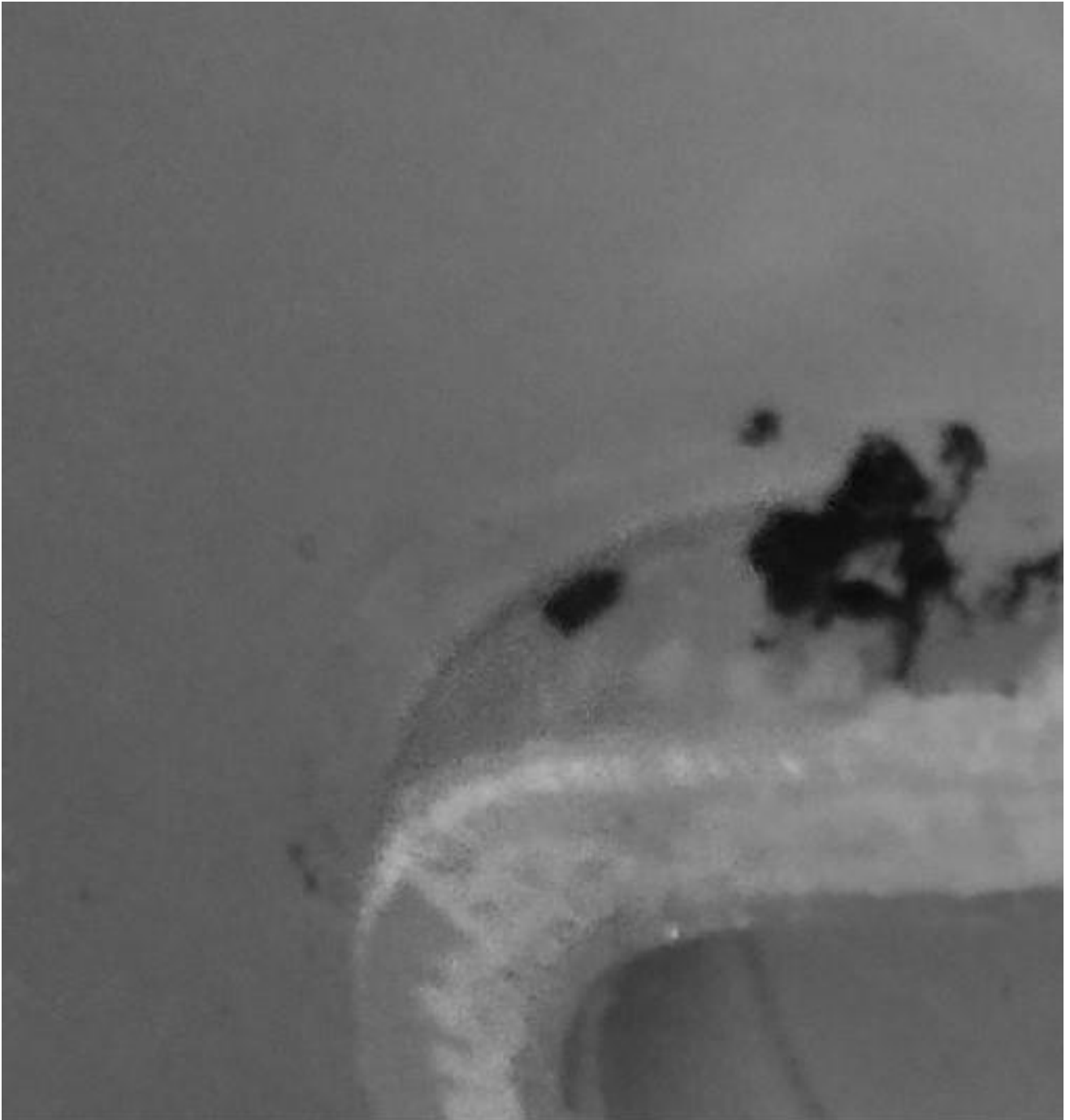
RESULTS



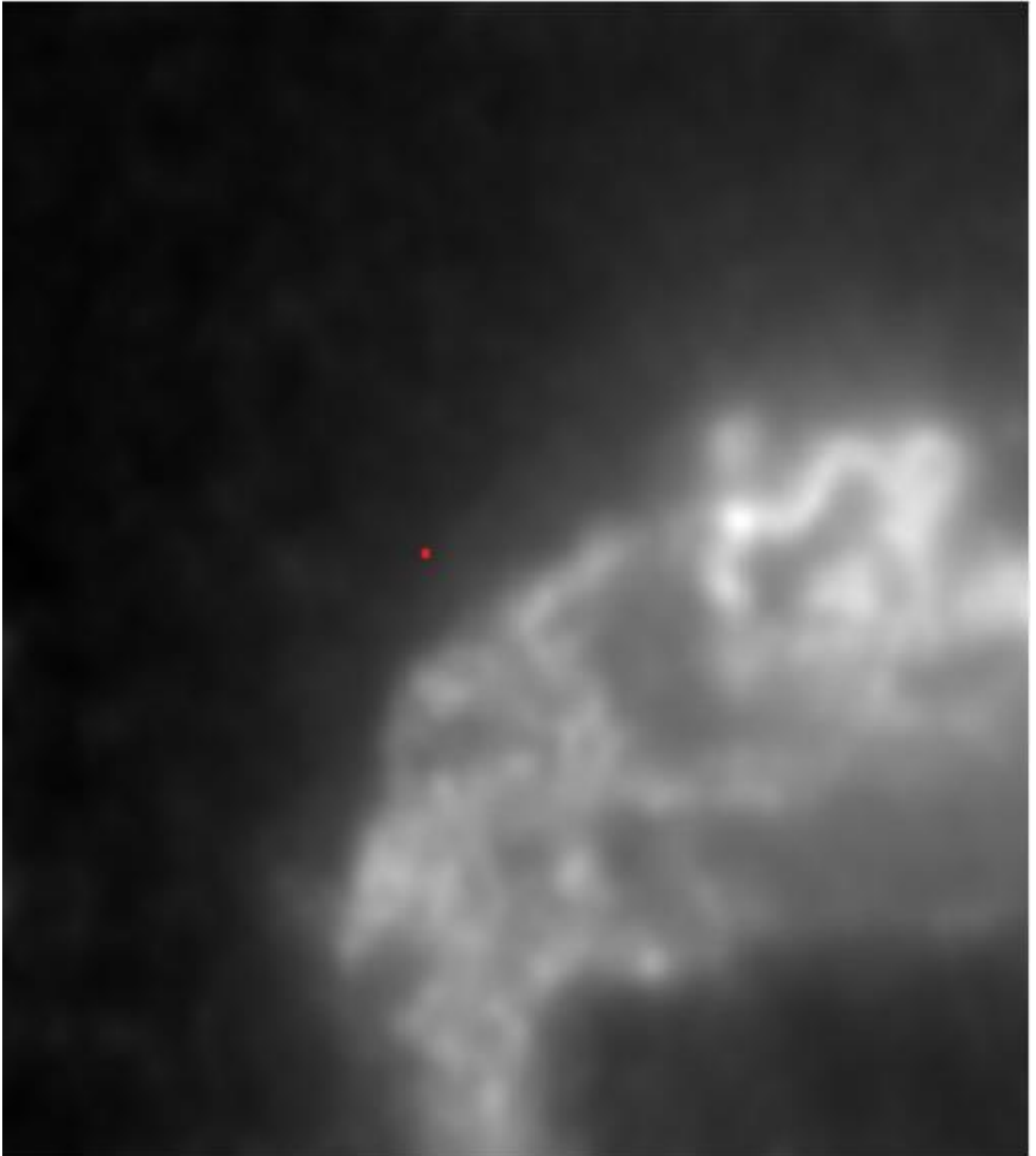
DYE FIXED AT UPPER SECTION



SECTION OF A MOULD WHERE EVA MATERIAL IS HANGING AFTER REMOVAL OF PREPARED SHOE FROM MOULD



**GREY PICTURE OF SECTION OF A MOULD WHERE EVA MATERIAL IS HANGING AFTER
REMOVAL OF PREPARED SHOE FROM MOULD**



OUTPUT OF K- MEANS