DAFTAR PUSTAKA

Ariyanti, G. (2010, November). Dekomposisi Nilai Singular dan Aplikasinya. Seminar Nasional Matematika dan Pendidikan Matematika, Yogyakarta.

Atmajaya, I.M. (2018) Kompresi Citra Medis Menggunakan *Packet Wavelet Transform* Dan *Run Length Encoding* Jurnal Matrix, Vol.8 No. 1

Eswaraiah, R., & Reddy, E. S. (2014). *A Fragile ROI-Based Medical Image Watermarking Technique with Tamper Detection and Recovery*. 2014 Fourth International Conference on Communication Systems and Network Technologies

Fikri, M., Samsurizal, Nurmiati Pasra. (2021). Singular Value Decomposition (SVD) Berdasarkan Intensitas Pencahayaannya Untuk Pengenal Wajah.Jurnal Ilmiah Setrum Vol.10 No.1 6-14

Fitriah, K. & Hakim.. Segmentasi Region Of Interest (ROI) Garis Telapak Tangan Menggunakan Deteksi Tepi Sobel.Jurnal Explore IT, Vol. 11 No. 1 http://jurnal.yudharta.ac.id/v2/index.php/EXPLORE-IT/

Harjito, B., Anisah, N., Suryani E.(2015). Digital Watermarking Image dengan Menggunakan Discrete Wavelet Transform dan Singular Value Decomposition (DWT-SVD) untuk Copyright Labeling. Jurnal IT smart, Vol 4. No 1

Jung, H.(2021). Basic Physical Principles and Clinical Applications of Computed Tomography. Korean Society Of Medical Physics. https://doi.org/10.14316/pmp.2021.32.1.1

Mathworks. Diakses pada Juli 2023. https://www.mathworks.com/matlabcentral/

Munir, R .(2020). Singular Value Decomposition. Diakses pada 22 Juli 2023. https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-19b-Singular-value-decomposition.pdf

Munir, R. (2006, September). Sekilas Image Watermaking untuk memproteksi Citra Digital dan Aplikasinya pada Citra Medis. Seminar Indonesian Conference on Telecommunications (ICTEL), Bandung.

NIH Nasional Library of Medacine. Diakses pada 27 juli 2023. https://medpix.nlm.nih.gov/

Nilai Eigen dan Vektor eigen. Diakses pada 26 Juli 2023. https://repository.dinus.ac.id/docs/ajar/Nilai_Eigen_dan_Vektor_Eigen.pdf

Putra, D. (2010). Pengolahan Citra Digital. Andi, Yogyakarta

Rayachoti, E., & Edara, S. R. (2014). *A novel medical image watermarking technique for detecting tampers inside ROI and recovering original ROI.* 2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)

Sulistiyanti, S.R., Fx Arianto Setyawan, Muhammad Kamaruddin.(2016). Pengolahan Citra Dasar dan Contoh Penerapannya.Yogyakarta:Teknosain.

Tareef, A., Al-Ani, A., Hung Nguyen, & Yuk Ying Chung. (2014). *A novel tamper detection-recovery and watermarking system for medical image authentication and EPR hiding.* 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.

Utari, C.T. (2016). Implementasi Algoritma *Run Length Encoding* Untuk Perancanganaplikasi Kompresi Dan Dekompresi File Citra. Jurnal TIMES, Vol. V No 2:24-31

LAMPIRAN

Lampiran 1

Kode pemrograman penyisipan watermark

```
%pemisahan ROI dan RONI
I=imread('medical_image.jpg');
I=rgb2gray(I);

imshow(I);
BW=roipoly;
BW=uint8(BW);


[r c]=size(BW);
for i=1:r
    for j=1:c
        if(BW(i,j)==0)
            BW(i,j)=1;
        else
            BW(i,j)=255;
        end
    end
end


subplot(1,3,1);
imshow(I);
title('Original Image');
subplot(1,3,2);
RONI=I./BW;
imshow(RONI);
title('RONI');
subplot(1,3,3);
ROI=I-RONI;
imshow(ROI);
title('ROI');
subplot(1,3,1);
imshow(I);
title('Original Image');
subplot(1,3,2);
RONI=I./BW;
imshow(RONI);
title('RONI');
subplot(1,3,3);
ROI=I-RONI;
imshow(ROI);
title('ROI');
imwrite(ROI, 'ROI.jpg');
imwrite(RONI, 'RONI.jpg');

I=im2double(I);

%kompresi
%watermark = rle_compression(ROI);
watermark = (ROI);
```

```
imwrite(watermark, 'watermark.jpg');
disp('watermark berhasil dibuat');

%penyisipan
watermarkedRONI = svd_embedding(RONI, watermark);
imwrite(watermarkedRONI, 'RONI berwatermark.jpg');

ROI = im2double(ROI);
watermarkedImage= watermarkedRONI+ ROI;
imshow(watermarkedImage);
title('citra berwatermark');
imwrite(watermarkedImage, 'citra berwatermark.jpg');

disp('penyisipan berhasil');


psnrValue = psnr(RONI,watermarkedRONI);

disp(['PSNR Value: ', num2str(psnrValue)]);

%fungsi kompresi
function compressed = rle_compression(ROI)

    roi_1d = ROI(:);

    count = 1;
    prev_value = roi_1d(1);
    compressed = zeros(1, numel(roi_1d));


    for i = 2:numel(roi_1d)
        if roi_1d(i) == prev_value
            count = count + 1;
        else
            compressed(i-1) = count;
            count = 1;
            prev_value = roi_1d(i);
        end
    end

    % Add the last run to the compressed data
    compressed(end) = count;
    compressed = [roi_1d(1), compressed(compressed > 0)];
end


% fungsi untuk penyisipan watermarking dengan metode SVD
function watermarkedRONI = svd_embedding(RONI, watermark)

RONI= im2double(RONI);
%RONI=imresize(RONI,[256 256]);
 [U, S, V] = svd(RONI);

      watermark=imresize(watermark,[256 256]);
     watermarkedS = S;
     alfa= input('The alfa Value = ');
```

```
   [x y]=size(watermark);
   watermark = im2double(watermark);
      for i=1:x
       for j=1:y
          watermarkedS(i,j) = watermarkedS(i,j) + alfa *
watermark(i,j);
       end
   end

    [Uw, Sw, Vw]= svd(watermarkedS);

    watermarkedRONI = U * Sw * V';
end
```

Lampiran 2

Kode pemrograman verifikasi (ekstraksi dan pembandingan) watermark

```
% Ekstraksi watermark dan pembandingan
%Ekstraksi
watermarkedImage = imread('citra_berwatermark.jpg');
watermarkedImage = im2double(watermarkedImage);

I_ROI = watermarkedImage(roi_start_y:roi_start_y+roi_height-1,
roi_start_x:roi_start_x+roi_width-1);
I_RONI = watermarkedImage(roni_start_y:roni_start_y+roni_height-1,
roni_start_x:roni_start_x+roni_width-1);

extractedWatermark = svd_extraction(I_RONI);

extractedROI = rle_decompression(extractedWatermark);

%Fungsi untuk dekompresi
function decompressed = rle_decompression(compressed)

    decompressed = [];

 i = 1;
    while i <= numel(compressed)
        value = compressed(i);
        count = round(compressed(i + 1));
        decompressed = [decompressed, ones(1, count) * value];
        i = i + 2;
    end

    if mod(numel(decompressed), 2) == 1
        decompressed = decompressed(1:end-1);
    end

    % Convert the decompressed data back to the original shape
    decompressed = reshape(decompressed, [],
numel(decompressed)/2);
end
```

```
% Pembandingan
extractedROI_resized = imresize(extractedROI, [roi_height,
roi_width]);
```

Lampiran 3

Kode pemrograman pengukuran kualitas

```
%Pengukuran kualitas
% Calculate the similarity metric (e.g., Normalized Correlation)
between ROI and I'ROI
similarity = calculate_similarity(ROI, extractedROI_resized);
% Step 6: Sensor restoration
% If the images are not the same (damage detected), use pixels
from IROI to restore I'ROI
if similarity < 0.95
    % Perform sensor restoration using pixels from IROI
    I_RONI = I_ROI;
    watermarkedImage(roni_start_y:roni_start_y+roni_height-1,
roni_start_x:roni_start_x+roni_width-1) = I_RONI;
end

% Menghitung PSNR dan NC
% PSNR
psnrValue = psnr(originalImage, watermarkedImage);

% NC
ncValue = calculate_similarity(ROI, I_ROI);


disp(['PSNR Value: ', num2str(psnrValue)]);
disp(['Normalized Correlation (NC) Value: ', num2str(ncValue)]);

% Fungsi pengukuran kualitas
function similarity = calculate_similarity(ROI1, ROI2)
    % Implement image similarity calculation here
    % Calculate the Normalized Correlation between ROI1 and ROI2
    mean1 = mean(ROI1(:));
    mean2 = mean(ROI2(:));
    numerator = sum((ROI1(:) - mean1) .* (ROI2(:) - mean2));
    denominator = sqrt(sum((ROI1(:) - mean1).^2) * sum((ROI2(:) -
mean2).^2));
    similarity = numerator / denominator;
end
```