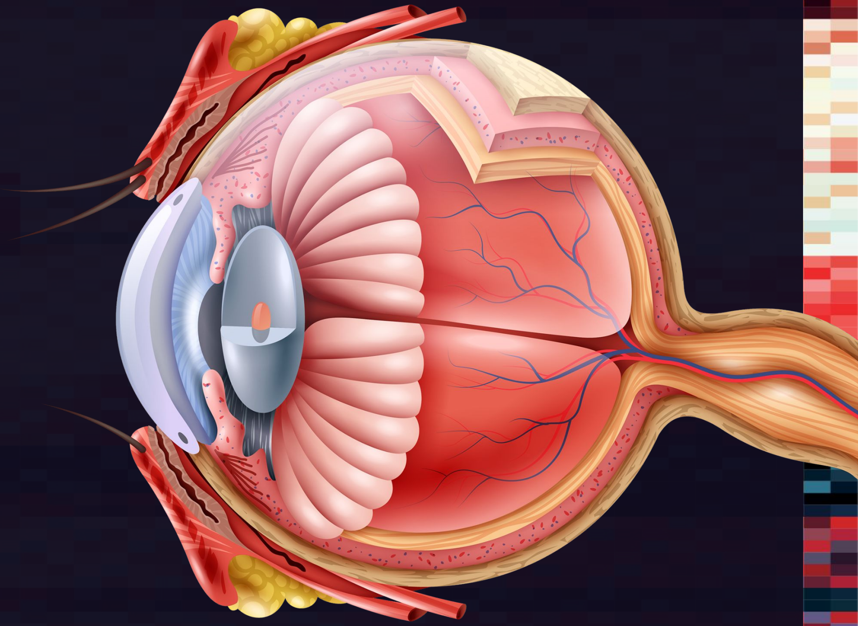
Laporan Proyek Akhir TD PCD 2022

Segmentasi Pembuluh Darah Retina dengan Metode Isotropic Undecimated Wavelet Transform (IUWT)

Sheinna Yendri - 6025212005



Contents lists available at SciVerse ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

Automatic wavelet-based retinal blood vessels segmentation and vessel diameter estimation

Abdolhossein Fathi*, Ahmad Reza Naghsh-Nilchi[1]

Department of Computer Engineering, The University of Isfahan, Isfahan, Iran

Menggunakan metode Continuous Wavelet Transform

Dataset: DRIVE & STARE

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

Enhancing retinal blood vessel segmentation in medical images using combined segmentation modes extracted by DBSCAN and morphological reconstruction

Kamran Mardani[a], Keivan Maghooli[b,*]

[a] Department of Biomedical Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran
[b] Department of Biomedical Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

Menggunakan metode DBSCAN & morphological reconstruction

Dataset: DRIVE, STARE, VAMPIRE

Pokok Bahasan

01 Pendahuluan

02 Dataset

03 Metodologi

04 Uji Coba dan Evaluasi

05 Kesimpulan

01 Pendahuluan

Pendahuluan

Perkembangan pembuluh darah pada retina atau yang biasa disebut neovaskularisasi merupakan suatu keadaan abnormal yang dapat berakibat pada pendarahan hingga kebutaan apabila tidak dideteksi sejak dini. Oleh sebab itu, penting untuk dapat melakukan segmentasi pembuluh darah mata yang nantinya dapat diklasifikasi untuk dideteksi sebagai pembuluh darah normal atau abnormal

02

Dataset

DRIVE

20 retinal images + ground truth

Pemilihan dataset DRIVE image 01-20 adalah agar performa segmentasi dapat dibandingkan dengan kedua paper.

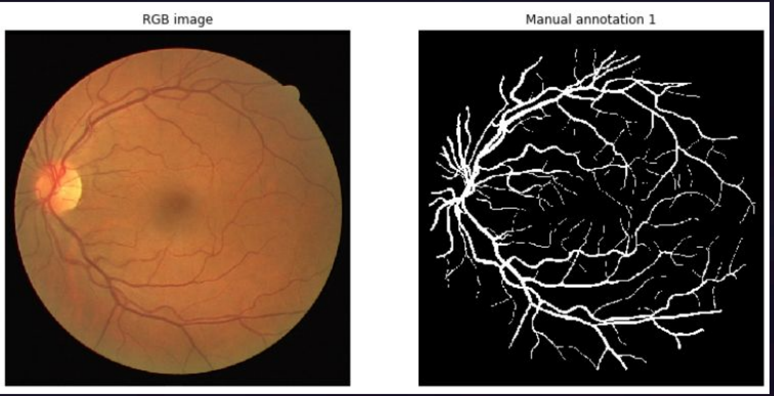


Table 4
Performance measures for DRIVE database after post-processing.

Image	ACC	SPE	SEN	AUC	Combination modes
02	0.9553	0.9729	0.8070	0.8900	(10, 2)&(10, 4)
07	0.9561	0.9709	0.7739	0.8724	(10, 2)&(10, 4)
08	0.9536	0.9704	0.7304	0.8504	(10, 3)&(10, 5)
11	0.9511	0.9685	0.7695	0.8690	(10, 2)&(10, 4)
13	0.9378	0.9536	0.8053	0.8794	(10, 2)&(10, 4)
15	0.9519	0.9688	0.7493	0.8591	(10, 2)&(10, 4)
18	0.9532	0.9744	0.7541	0.8643	(10, 2)&(10, 4)
20	0.9562	0.9739	0.7441	0.8567	(10, 2)&(10, 4)
ALL	**0.9519**	**0.9692**	**0.7667**	**0.8676**	

Table 1
The obtained performances of all images on the both DRIVE and STARE databases.

Image no.	DRIVE			
	Sensitivity	Specificity	Predictive	Accuracy
1	0.8426	0.9720	0.7469	0.9605
2	0.7758	0.9828	0.8372	0.9616
3	0.7127	0.9770	0.7744	0.9507
4	0.7419	0.9804	0.7935	0.9585
5	0.7439	0.9821	0.8111	0.9598
6	0.7019	0.9820	0.8079	0.9547
7	0.6997	0.9833	0.8083	0.9574
8	0.7052	0.9783	0.7536	0.9548
9	0.7770	0.9725	0.7134	0.9566
10	0.7648	0.9808	0.7810	0.9630
11	0.7525	0.9734	0.7356	0.9536
12	0.7869	0.9725	0.7300	0.9565
13	0.7426	0.9727	0.7465	0.9502
14	0.8241	0.9660	0.6808	0.9545
15	0.7969	0.9786	0.7413	0.9656
16	0.7976	0.9763	0.7697	0.9602
17	0.7771	0.9743	0.7357	0.9576
18	0.8533	0.9669	0.6892	0.9579
19	0.8562	0.9807	0.8007	0.9704
20	0.8693	0.9645	0.6606	0.9575
Average	0.7768	0.9759	0.7559	0.9581

03 Metodologi

Rancangan Metode

1

Thresholding

RGB → grayscale, implementasi thresholding

2

Shrinking

Menggunakan morfologi erosi untuk memperbaiki FOV

3

Segmentasi

Menggunakan metode Isotropic Undecimated Wavelet Transform (IUWT)

4

Cleaning

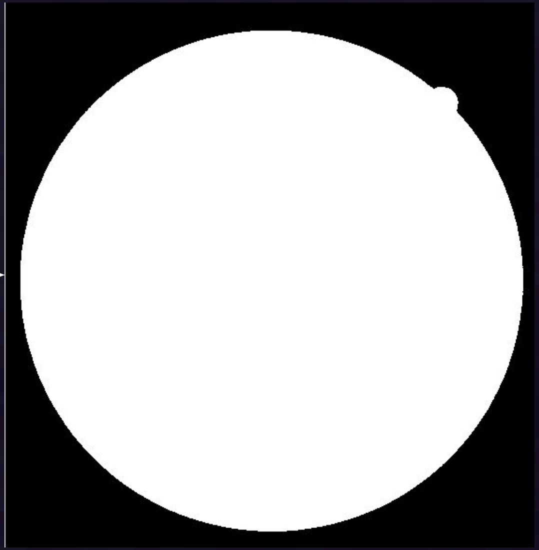
Menghilangkan noise atau objek kecil dan mengisi lubang kecil

1. Thresholding

- Mengubah citra RGB menjadi citra grayscale.
- Menentukan nilai threshold.
- Untuk citra DRIVE, digunakan nilai threshold = 0.13.
- Intensitas pixel < threshold → background (hitam)
- Intensitas pixel > threshold → object retina (putih)

```
%read file input
img = uigetfile('*.bmp;*.ppm;*.tif');
im = imread(img);
im = rgb2gray(im);

%masking input image
bw_mask = mask(im);
```

```
function final_mask=mask(im)
bw_mask = im2bw(im, 0.13);
% figure, imshow(bw_mask);
d1 = max(sum(bw_mask));
d2 = max(sum(bw_mask, 2));
d = max(d1, d2);
siz = max(5, round(d * .025));
final_mask = imerode(bw_mask, ones(siz));
% figure,imshow(final_mask);
imwrite(final_mask,'mask.bmp');
```



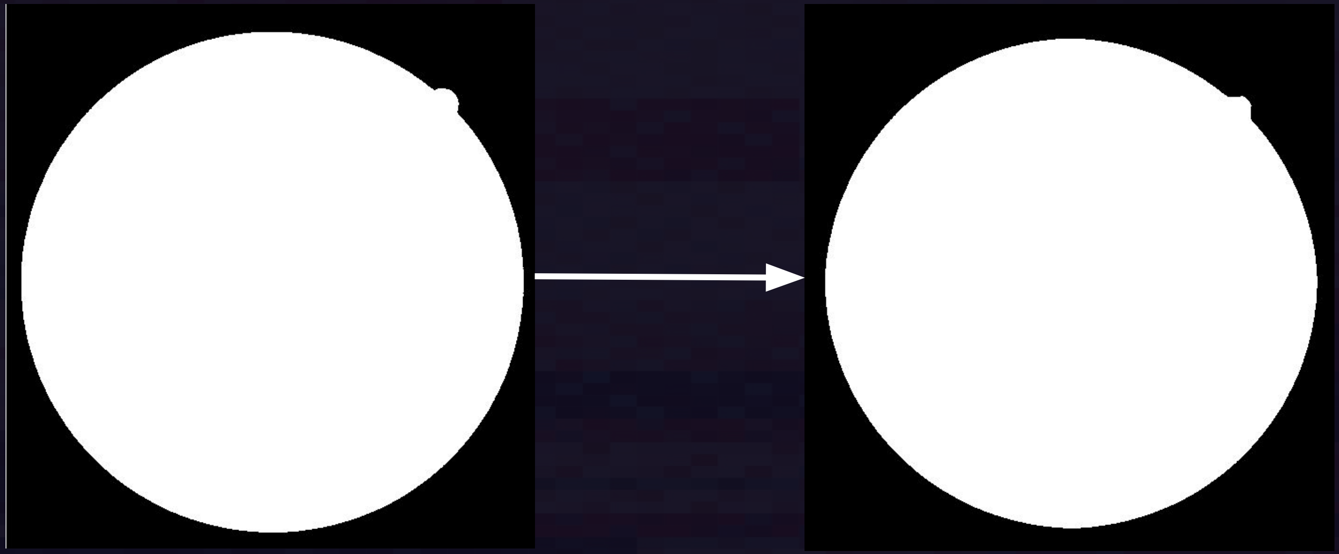
Thresholding

2. Shrinking

- Penyusutan binary image untuk memperbaiki field of view (FOV).
- Dilakukan dengan menggunakan operasi morfologi erosi.
- Elemen penstruktur: matriks dua dimensi NxN.

```
%read file input
img = uigetfile('*.bmp;*.ppm;*.tif');
im = imread(img);
im = rgb2gray(im);

%masking input image
bw_mask = mask(im);
```

```
function final_mask=mask(im)
bw_mask = im2bw(im, 0.13);
% figure, imshow(bw_mask);
d1 = max(sum(bw_mask));
d2 = max(sum(bw_mask, 2));
d = max(d1, d2);
siz = max(5, round(d * .025));
final_mask = imerode(bw_mask, ones(siz));
% figure,imshow(final_mask);
imwrite(final_mask,'mask.bmp');
```



Shrinking

3. Segmentasi

A. WAVELET TRANSFORM

- Menggunakan metode Isotropic Undecimated Wavelet Transform.
- IUWT mempertahankan konstruksi filter bank pada bi-orthogonal wavelet transform (DWT) dengan mengeliminasi tahap penipisan (decimation).
- Untuk citra DRIVE, digunakan level wavelet pada level 2 dan 3.
- Input: citra grayscale retina dan citra mask retina.
- Output: citra keabuan yang memperjelas bagian citra yang berupa garis yaitu bagian pembuluh darah retina.

```
%default value for segmentation
iuwt_dark = true;
iuwt_inpainting = false;
iuwt_w_levels  = 2:3;
iuwt_w_thresh  = 0.2;
iuwt_px_remove = 0.05;
iuwt_px_fill   = 0.05;

%make wavelet & threshold image
[bw_wavelet, iuwt_dark, iuwt_w_levels, iuwt_w_thresh] = dlg_iuwt(...
                im, bw_mask, iuwt_dark, iuwt_w_levels, iuwt_w_thresh, iuwt_inpainting);
```

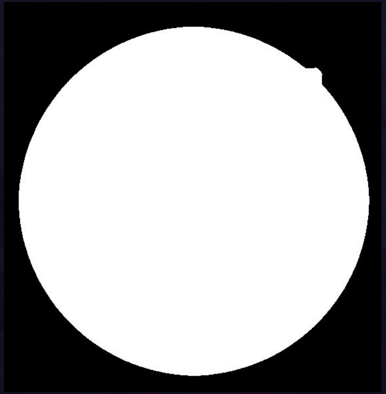
```
% Compute and store full wavelet transform
handles.w_full = iuwt_vessel_all(im, handles.WAVELET_LEVELS);
```

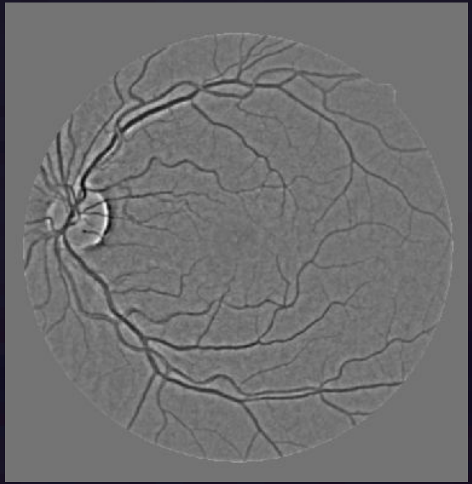
```
function [w, s_out] = iuwt_vessel_all(im, levels, padding)
% Compute transform
for ii = 1:levels(end)
    % Create convolution kernel
    h = dilate_wavelet_kernel(b3, 2^(ii-1)-1);

    % Convolve and subtract to get wavelet level
    s_out = imfilter(s_in, h' * h, padding);

    % Store wavelet level only if it's in LEVELS
    ind = find(levels == ii);
    if isscalar(ind)
        w(:,:,ind) = s_in - s_out;
    end

    % Update input for new iteration
    s_in = s_out;
end

% Remove singleton dimensions
w = squeeze(w);
```

```
function h2 = dilate_wavelet_kernel(h, spacing)
% Dilates a wavelet kernel by entering SPACING zeros between each
% coefficient of the filter kernel H.

% Check input
if ~isvector(h) && ~isscalar(spacing)
    error(['Invalid input to DILATE_WAVELET_KERNEL: ' ...
        'H must be a vector and SPACING must be a scalar']);
end

% Preallocate the expanded filter
h2 = zeros(1, numel(h) + spacing * (numel(h) - 1));
% Ensure output kernel orientation is the same
if size(h,1) > size(h,2)
    h2 = h2';
end
% Put in the coefficients
h2(1:spacing+1:end) = h;
```

```
function [w, s_out] = iuwt_vessel_all(im, levels, padding)
% Compute transform
for ii = 1:levels(end)
    % Create convolution kernel
    h = dilate_wavelet_kernel(b3, 2^(ii-1)-1);

    % Convolve and subtract to get wavelet level
    s_out = imfilter(s_in, h' * h, padding);

    % Store wavelet level only if it's in LEVELS
    ind = find(levels == ii);
    if isscalar(ind)
        w(:,:,ind) = s_in - s_out;
    end

    % Update input for new iteration
    s_in = s_out;
end

% Remove singleton dimensions
w = squeeze(w);
```

```
% Preallocate wavelet output; 3-d even if input is a vector
w = zeros([size(im) length(levels)], wclass);

% B3 spline coefficients for filter
b3 = [1 4 6 4 1] / 16;
```





Wavelet Transform

3. Segmentasi

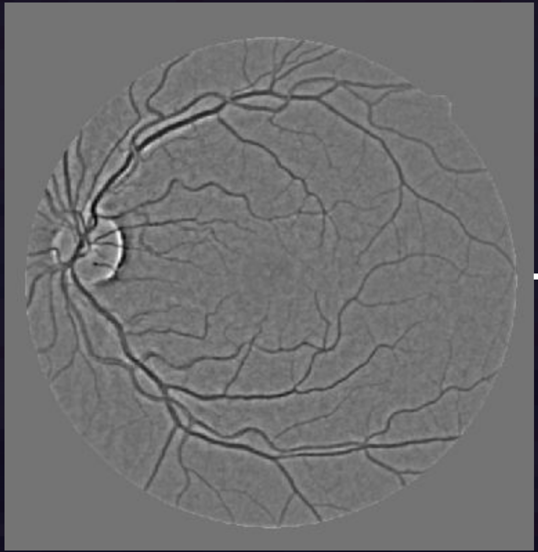
B. THRESHOLDING

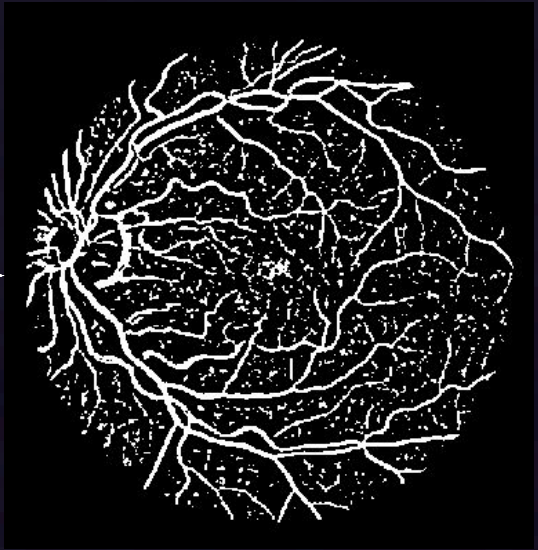
- Dilakukan thresholding lagi, tetapi kali ini untuk menentukan persentase pixel yang dianggap sebagai elemen pembuluh darah ditandai dengan warna putih.
- Untuk citra DRIVE, digunakan threshold untuk elemen vaskular sebesar 20-25%.
- Selain menentukan threshold, perlu juga menentukan apakah pixel vaskuler dari citra hasil wavelet transform lebih gelap dibandingkan pixel lainnya.
    - Jika lebih gelap: pixel dengan intensitas rendah → elemen vaskular.
    - Jika lebih terang: pixel dengan intensitas tinggi → elemen vaskular.

```matlab
function update_thresholded_image(hObject)
handles = guidata(hObject);
[handles.bw, handles.sorted_pix] = ...
        percentage_segment(handles.w, ...
                get_threshold_scale(handles), ...
                get_dark_vessels(handles), ...
                handles.bw_mask, ...
                handles.sorted_pix);
guidata(hObject, handles);
```

```matlab
function [bw, sorted_pix] = percentage_segment(im, proportion, dark, bw_mask, sorted_pix)
if dark
    proportion = 1 - proportion;
end
[threshold, sorted_pix] = percentage_threshold(sorted_pix, proportion, true);
if dark
    bw = im <= threshold;
else
    bw = im > threshold;
end
if ~isempty(bw_mask)
    bw = bw & bw_mask;
end
```

```matlab
function [threshold, data_sorted] = percentage_threshold(data, proportion, sorted)
proportion = 1-proportion;
thresh_ind = round(proportion * numel(data_sorted));
if thresh_ind > numel(data_sorted)
    threshold = Inf;
elseif thresh_ind < 1
    threshold = -Inf;
else
    threshold = data_sorted(thresh_ind);
end
```





Segmentation - Thresholding

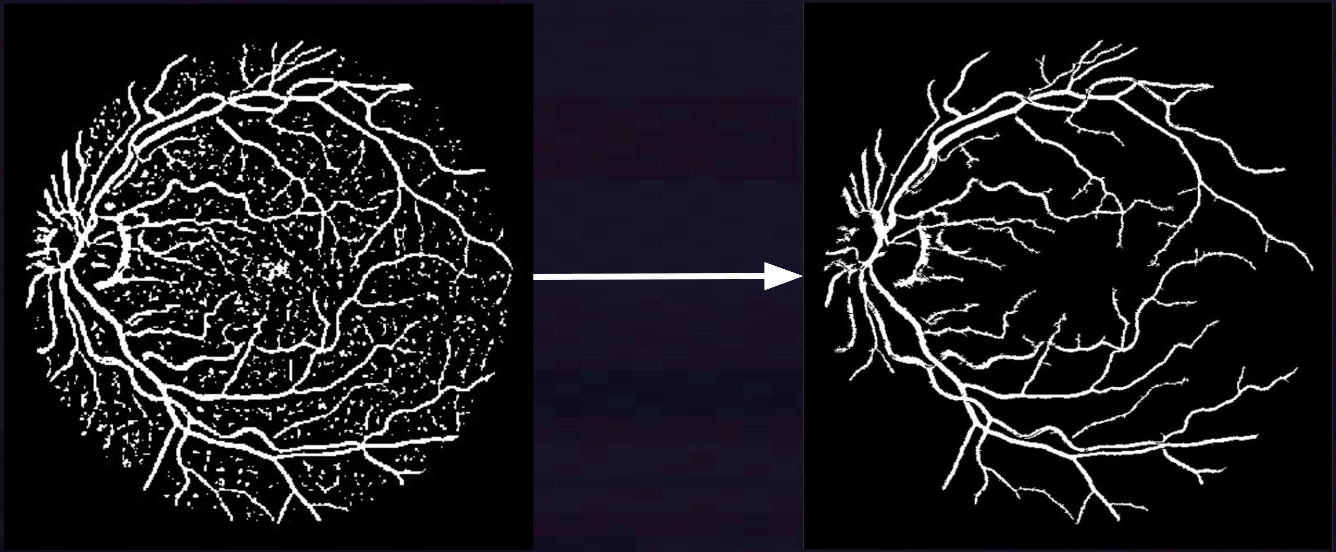
4. Cleaning

- Setelah dilakukan thresholding, terlihat citra hasil segmentasi masih memiliki banyak noise dan segmen yang terpisah-pisah.
- Tahap cleaning bertujuan untuk menghapus objek-objek kecil yang dapat disebut sebagai noise (*min_object_size*), dan juga mengisi lubang-lubang kecil antar segmen yang terputus/terpisah (*min_holes_size*).
- Untuk citra DRIVE, nilai min_object_size=0.05% dan min_holes_size=0.05%.
- Proses cleaning:
    - Jika luas area objek < min_object_size → akan dihapus.
    - Jika luas area lubang < min_holes_size → akan diisi.

```
%make segmented
[bw_segmented, iuwt_px_remove, iuwt_px_fill] = dlg_clean_segmented(...
            bw_wavelet, iuwt_px_remove, iuwt_px_fill, bw_mask);
```

```
% Create updated binary image
handles.bw = clean_segmented_image(handles.bw_orig, min_object_size, min_hole_size);
```

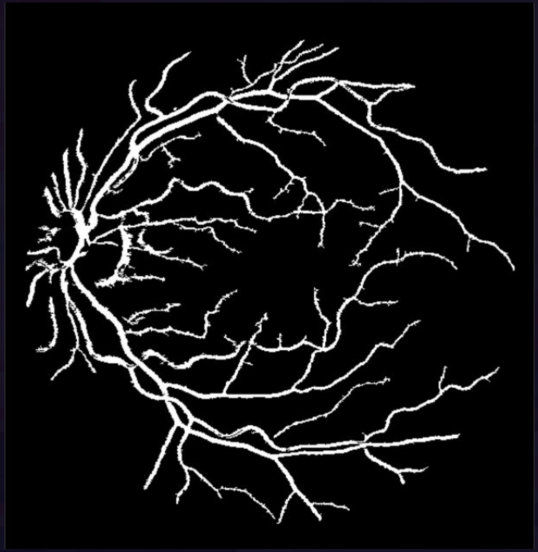
```
function bw_clean = clean_segmented_image(bw, min_object_size, min_hole_size)

% Remove small objects, if necessary
if min_object_size > 0
    cc_objects = bwconncomp(bw);
    area_objects = cellfun('size', cc_objects.PixelIdxList, 1);
    bw_clean = false(size(bw));
    inds = area_objects >= min_object_size;
    bw_clean(cell2mat(cc_objects.PixelIdxList(inds)')) = true;
else
    bw_clean = bw;
end
% Fill in holes, if necessary
if min_hole_size > 0
    cc_holes = bwconncomp(~bw_clean);
    area_holes = cellfun('size', cc_holes.PixelIdxList, 1);
    inds = area_holes < min_hole_size;
    bw_clean(cell2mat(cc_holes.PixelIdxList(inds)')) = true;
end
```



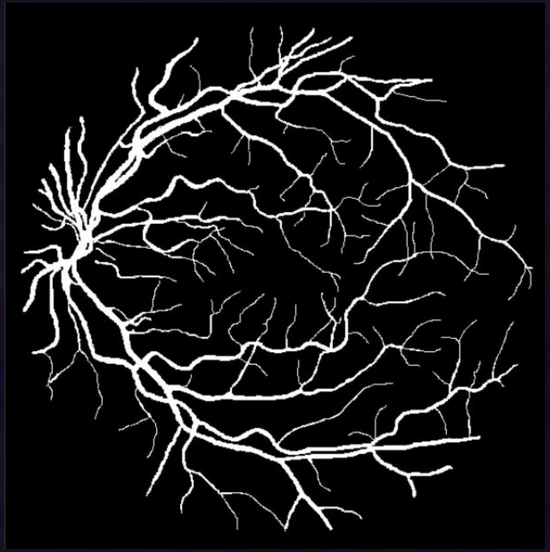
Cleaning

image_01 from DRIVE dataset

Original image



Segmentation Result



Ground Truth

04

Uji Coba & Evaluasi

Skenario Uji Coba

UJI HASIL SEGMENTASI

Evaluasi nilai accuracy, specificity, sensitivity, precision, dan area under the curve antara hasil segmentasi dengan ground truth yang disediakan oleh dataset DRIVE, mengikuti formula berikut.

Accuracy (Acc) = (TP + TN) / (TP + TN + FP + FN)

Specificity (Sp) = TN / (FP + TN)

Sensitivity (Se) = TP / (TP + FN)

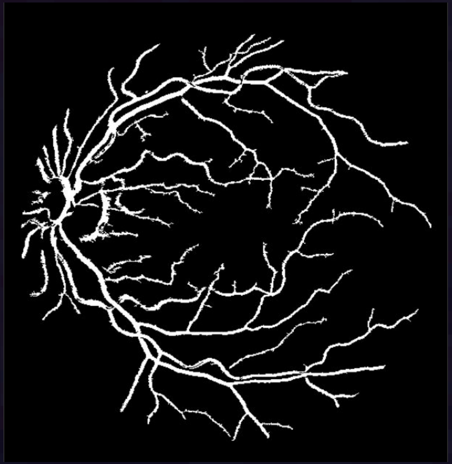
Precision (Pre) = TP / (TP + FP)

Area under the curve (AUC) = (Se + Sp) / 2

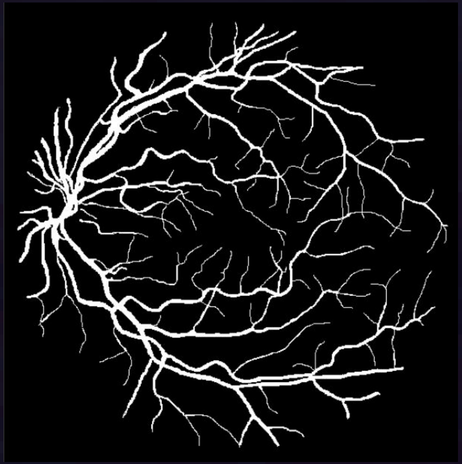
Hasil Uji Coba (Visual)

image_01 from DRIVE dataset: normal retina

Original image



Segmentation Result

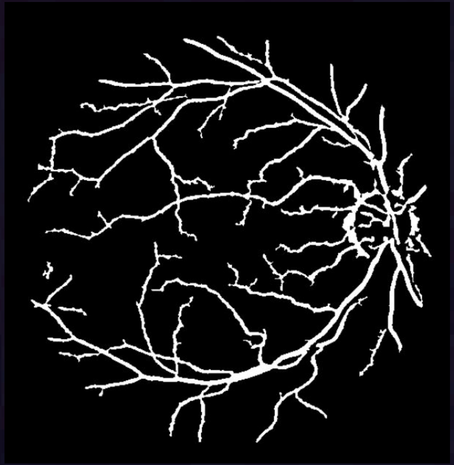


Ground Truth

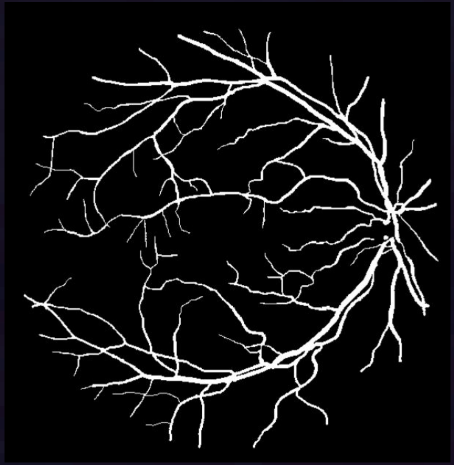
Hasil Uji Coba (Visual)

image_20 from DRIVE dataset: normal retina

Original image



Segmentation Result

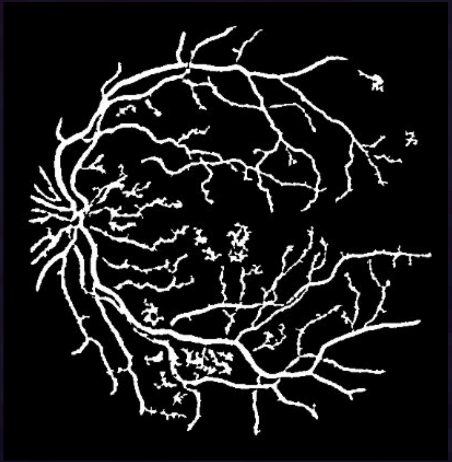


Ground Truth

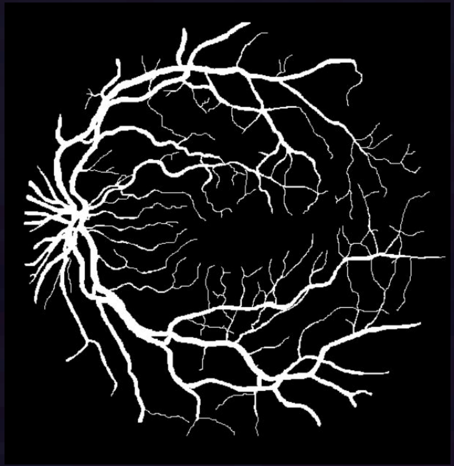
Hasil Uji Coba (Visual)

image_03 from DRIVE dataset: background diabetic retinopathy

Original image



Segmentation Result

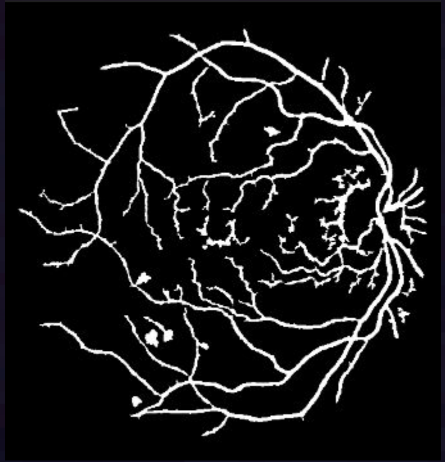


Ground Truth

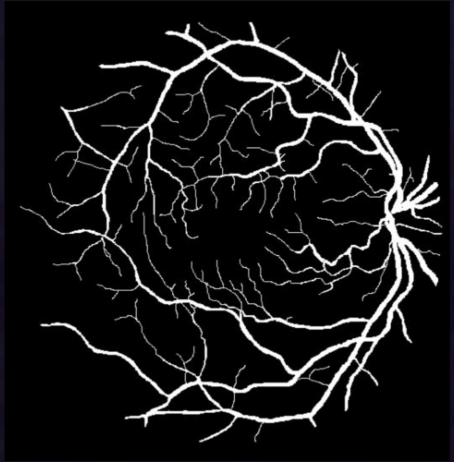
Hasil Uji Coba (Visual)

image_14 from DRIVE dataset: background diabetic retinopathy

Original image



Segmentation Result



Ground Truth

Evaluasi

Dilakukan pada 20 citra retina pada dataset DRIVE.

Rata-rata hasil evaluasi:

- Accuracy: 95.29%
- Specificity: 97.65%
- Precision: 74.85%
- Sensitivity: 77.95%
- AUC: 87.80%

DRIVE	Accuracy	Specificity	Precision	Sensitivity	AUC
image_01	0.9555	0.9719	0.7332	0.8328	0.90235
image_02	0.9548	0.9783	0.7926	0.7966	0.88745
image_03	0.9468	0.9786	0.7734	0.7201	0.84935
image_04	0.9514	0.975	0.7446	0.755	0.865
image_05	0.9527	0.9763	0.7597	0.767	0.87165
image_06	0.9515	0.9943	0.9119	0.7302	0.86225
image_07	0.9534	0.9933	0.8934	0.7376	0.86545
image_08	0.948	0.9892	0.8165	0.6892	0.8392
image_09	0.953	0.9752	0.7137	0.7594	0.8673
image_10	0.9538	0.974	0.7156	0.7732	0.8736
image_11	0.9488	0.9736	0.7216	0.7502	0.8619
image_12	0.9533	0.9744	0.7293	0.7835	0.87895
image_13	0.9474	0.9743	0.7467	0.7578	0.86605
image_14	0.9518	0.9695	0.6839	0.7955	0.8825
image_15	0.9517	0.9654	0.6329	0.8212	0.8933
image_16	0.9539	0.975	0.7464	0.7992	0.8871
image_17	0.9533	0.9755	0.7282	0.778	0.87675
image_18	0.9558	0.9715	0.6999	0.821	0.89625
image_19	0.9611	0.9726	0.7336	0.8679	0.92025
image_20	0.9597	0.9714	0.6927	0.8544	0.9129
Average	0.952885	0.976465	0.74849	0.77949	0.877978

DRIVE	Sensitivity	Specificity	Precision	Accuracy
image_01	0.8328	0.9719	0.7332	0.9555
image_02	0.7966	0.9783	0.7926	0.9548
image_03	0.7201	0.9786	0.7734	0.9468
image_04	0.755	0.975	0.7446	0.9514
image_05	0.767	0.9763	0.7597	0.9527
image_06	0.7302	0.9943	0.9119	0.9515
image_07	0.7376	0.9933	0.8934	0.9534
image_08	0.6892	0.9892	0.8165	0.948
image_09	0.7594	0.9752	0.7137	0.953
image_10	0.7732	0.974	0.7156	0.9538
image_11	0.7502	0.9736	0.7216	0.9488
image_12	0.7835	0.9744	0.7293	0.9533
image_13	0.7578	0.9743	0.7467	0.9474
image_14	0.7955	0.9695	0.6839	0.9518
image_15	0.8212	0.9654	0.6329	0.9517
image_16	0.7992	0.975	0.7464	0.9539
image_17	0.778	0.9755	0.7282	0.9533
image_18	0.821	0.9715	0.6999	0.9558
image_19	0.8679	0.9726	0.7336	0.9611
image_20	0.8544	0.9714	0.6927	0.9597
Average	0.77949	0.976465	0.74849	0.952885

PROPOSED METHOD

Table 1
The obtained performances of all images on the both DRIVE and STARE databases.

Image no.	DRIVE			
	Sensitivity	Specificity	Predictive	Accuracy
1	0.8426	0.9720	0.7469	0.9605
2	0.7758	0.9828	0.8372	0.9616
3	0.7127	0.9770	0.7744	0.9507
4	0.7419	0.9804	0.7935	0.9585
5	0.7439	0.9821	0.8111	0.9598
6	0.7019	0.9820	0.8079	0.9547
7	0.6997	0.9833	0.8083	0.9574
8	0.7052	0.9783	0.7536	0.9548
9	0.7770	0.9725	0.7134	0.9566
10	0.7648	0.9808	0.7810	0.9630
11	0.7525	0.9734	0.7356	0.9536
12	0.7869	0.9725	0.7300	0.9565
13	0.7426	0.9727	0.7465	0.9502
14	0.8241	0.9660	0.6808	0.9545
15	0.7969	0.9786	0.7413	0.9656
16	0.7976	0.9763	0.7697	0.9602
17	0.7771	0.9743	0.7357	0.9576
18	0.8533	0.9669	0.6892	0.9579
19	0.8562	0.9807	0.8007	0.9704
20	0.8693	0.9645	0.6606	0.9575
Average	0.7768	0.9759	0.7559	0.9581

PAPER 1: CONTINUOUS WAVELET TRANSFORM

DRIVE	ACC	SPE	SEN	AUC
image_02	0.9548	0.9783	0.7966	0.88745
image_07	0.9534	0.9933	0.7376	0.86545
image_08	0.948	0.9892	0.6892	0.8392
image_11	0.9488	0.9736	0.7502	0.8619
image_13	0.9474	0.9743	0.7578	0.86605
image_15	0.9517	0.9654	0.8212	0.8933
image_18	0.9558	0.9715	0.821	0.89625
image_20	0.9597	0.9714	0.8544	0.9129
Average	0.95245	0.977125	0.7785	0.877813

PROPOSED METHOD

Table 4
Performance measures for DRIVE database after post-processing.

Image	ACC	SPE	SEN	AUC	Combination modes
02	0.9553	0.9729	0.8070	0.8900	(10, 2)&(10, 4)
07	0.9561	0.9709	0.7739	0.8724	(10, 2)&(10, 4)
08	0.9536	0.9704	0.7304	0.8504	(10, 3)&(10, 5)
11	0.9511	0.9685	0.7695	0.8690	(10, 2)&(10, 4)
13	0.9378	0.9536	0.8053	0.8794	(10, 2)&(10, 4)
15	0.9519	0.9688	0.7493	0.8591	(10, 2)&(10, 4)
18	0.9532	0.9744	0.7541	0.8643	(10, 2)&(10, 4)
20	0.9562	0.9739	0.7441	0.8567	(10, 2)&(10, 4)
ALL	**0.9519**	**0.9692**	**0.7667**	**0.8676**	

PAPER 2: DBSCAN + MR

05 Kesimpulan

Kesimpulan

- Hasil segmentasi menggunakan metode IUWT pada citra retina berhasil mendapatkan nilai rata-rata accuracy 95.29%, specificity 97.65%, precision 74.85%, sensitivity 77.95%, dan AUC 87.80%.
- Dilakukan pengujian pada 20 citra retina dari dataset DRIVE dan dilakukan evaluasi hasil segmentasi antara metode yang diusulkan dan kedua paper, didapatkan:
    - Dibandingkan dengan paper pertama yaitu menggunakan metode continuous wavelet transform, didapatkan hasil segmentasi metode yang diusulkan memiliki nilai sensitivity dan specificity yang lebih baik.
    - Dibandingkan dengan paper kedua yaitu menggunakan metode DBSCAN dan MR, didapatkan hasil segmentasi metode yang diusulkan memiliki nilai accuracy, specificity, sensitivity, dan AUC yang lebih baik.

Terima kasih