Pseudo-code for Boundary Extraction for 2D binary image by Distance Transformation Algorithm

Developed by Astushi Imiya & Ken Tatara

Boundary Extraction by Distance Transformation

INPUT : $N \times N$ binary image $\{P(i,j)\}$.
/$P(i,j) = 0$ is a background pixel and $P(i,j) = 1$ is an object pixel /

OUTPUT : $N \times N$ binary image $B = B(i,j)$.
/B is a boundary of image./

/move image/

procedure **TRANCELATE-ON-ORTHOGONAL-DIRECTIONS**
begin
input $P(i,j)$
for $i = 0,1,2,...,N, j = 0,1,2,...,N$ **do**
if $P(i,j) = 1$
then $P_m(i,j) = P(i + move_x, j + move_y)$
/$move_x = -1,0,1, move_y = -1,0,1$/
output $\{P_m(i,j) = 1\}$
end

/extract inner and outer boundary/

procedure **BOUNDARY-EXTRACTION**
begin
input $P(i,j)$
input $P_m(i,j)$
for $i = 0,1,2,...,N, j = 0,1,2,...,N$ **do**
$B_{inner} = \bigcup\{P \backslash (P \cap P_m)\}$ /B_{inner} is INNER Boundary/
$B_{outer} = \bigcup\{(P \cap P_m) \backslash P\}$ /B_{outer} is OUTER Boundary/
output $\{B_{inner}\}$
output $\{B_{outer}\}$
end

/boundary extraction/

procedure **BOUNDARY-EXTRACTION**

begin

    begin

        TRANCELATE-ON-ORTHOGONAL-DIRECTIONS

    end

    begin

        BOUNDARY-EXTRACTION

    end

        output $B = B_{inner} \cup B_{outer}$

end