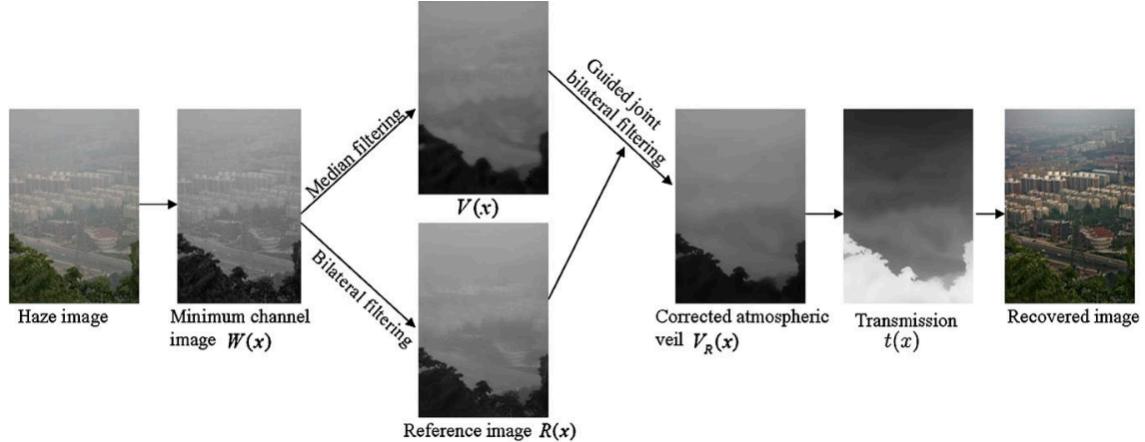


עבודה 2 עיבוד תמונה.

# מאמר 1

האלגוריתם:



נוסחאות

$$\vec{I}(x) = \vec{J}(x)t(x) + \vec{A}(1 - t(x))$$

נוסחה כללית

$$\vec{J}(x) = \frac{\vec{I}(x) - \vec{A}}{\max\{t(x), t_0\}} + \vec{A}$$

נוסחה לשczור התמונה

$$I_D^B(x) = \frac{\sum_{y \in \Omega(x)} f(x-y) * g(D(x) - D(y)) * I(y)}{\sum_{y \in \Omega(x)} f(x-y) * g(D(x) - D(y))}$$

נוסחה של ביליטרל פילטר

$$f(x-y) = \exp\left(-\frac{(x-y)^2}{2\sigma_s^2}\right)$$

$$g(D(x) - D(y)) = \exp\left(-\frac{(D(x) - D(y))^2}{2\sigma_r^2}\right)$$

$$h(x) = \exp\left(-\frac{x^2}{2\sigma_t^2}\right)$$

$$R(x) = \frac{\sum_{y \in \Omega(x)} f(x-y) * g(W(x) - W(y)) * W(y)}{\sum_{y \in \Omega(x)} f(x-y) * g(W(x) - W(y))}$$

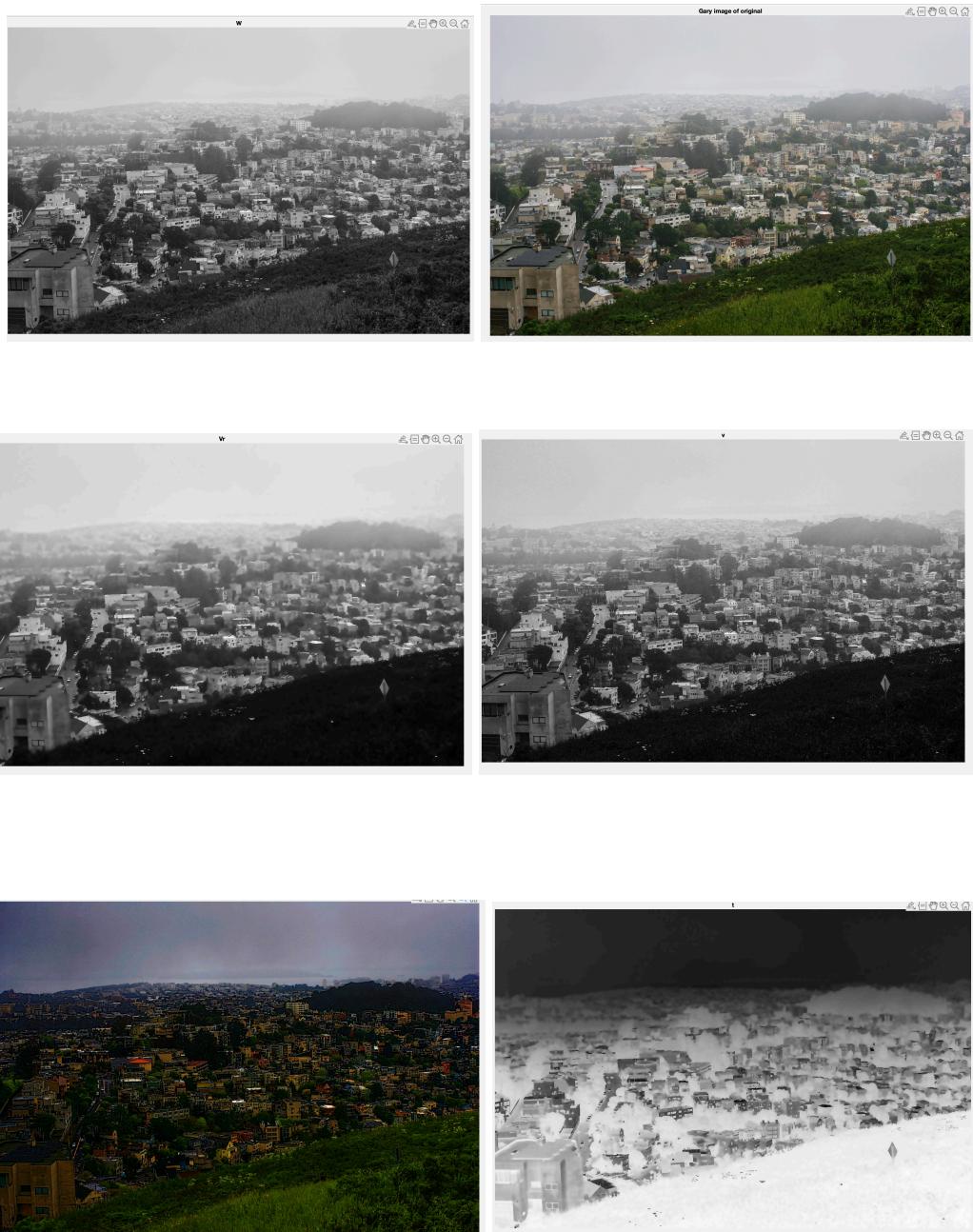
$$W(x) = \min_{c \in \{r,g,b\}} (I^c(x))$$

נוסחה לחישוב תמונה רפלקטיבית.

$$V_R(x) = \frac{\sum_{y \in \Omega(x)} f(x-y) * g(R(x) - R(y)) * h(V(y) - R(y)) * V(y)}{\sum_{y \in \Omega(x)} f(x-y) * g(R(x) - R(y)) * h(V(y) - R(y))}$$

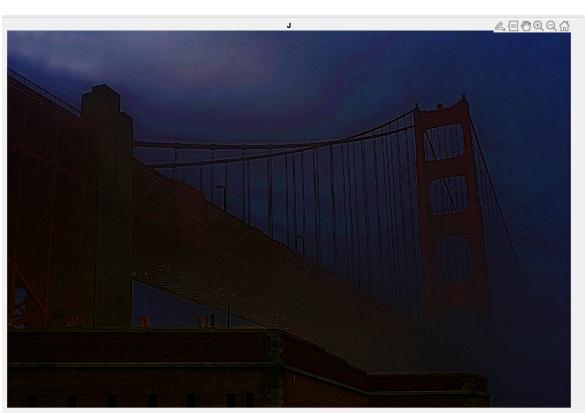
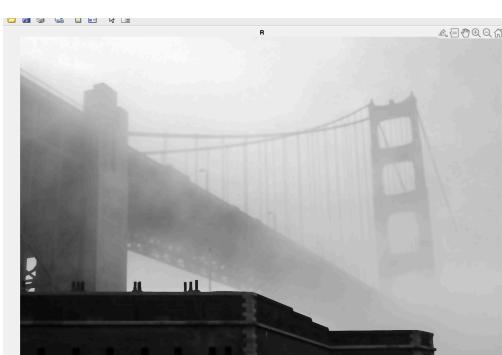
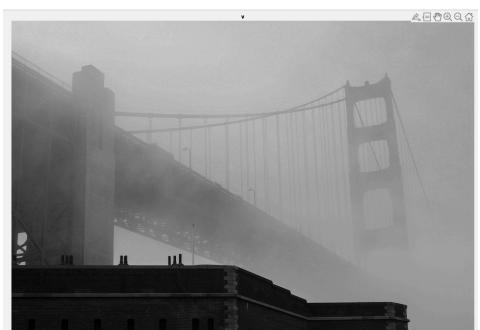
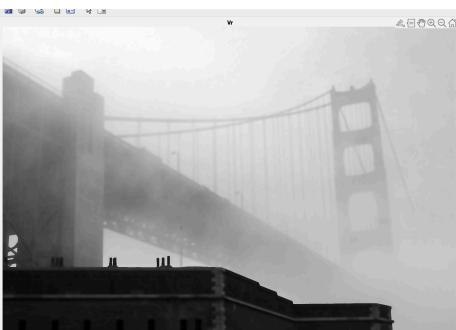
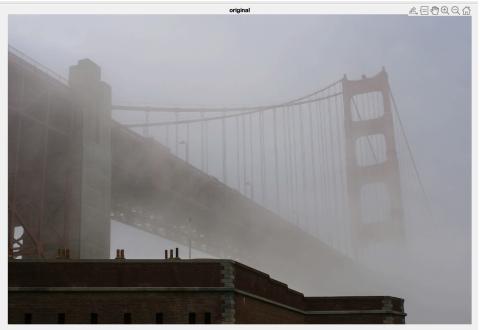
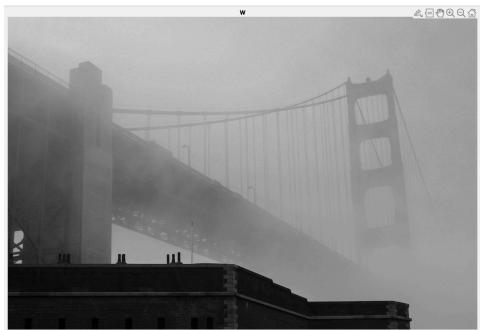
נוסחה ל- $\nabla$

## תוצאות



מימין לשמאל למטה: 1. תמונה מקורית. 2. מינימום dgr.  
3. אחרי מידיום פילטר בגודל 4. 4. אחרי ביליטרל פילטר  
5 t 5  
6. תמונה אחרי שחזור





תיקון

```
;clear all  
;  
;close all  
;  
;%  
;  
;('image=imread('foggysf2.jpg  
;('figure,imshow(image),title('original  
;%  
;([m,n,o]=size(image]  
;(  
;sigma_s=0.03*min(n,m  
;sigma_r=0.1  
;p=0.95  
;w=0.95  
;t0=0.25  
;Amin=240  
;OMEGA=10  
W & v %  
;W=zeros(m,n  
;b=W  
for i=1:m
```

```

        for j=1:n

            ;((:,w(i,j)=min(image(i,j

                end

            end

            ;('figure,imshow(uint8(w)),title('W

            ;('fprintf('W finished

            ;(w_double=double(w

            for i=OMEGA+1:m-OMEGA

                for j=OMEGA+1:n-OMEGA

                    ;(temp=w_double(i-OMEGA:i+OMEGA,j-OMEGA:j+OMEGA

                    ;((b(i,j)=median(median(temp

                        end

                    end

                    ;(c=b-median(abs(w-b),OMEGA

                    ;(v=max(min(p.*c,w),0

                    ;('figure,imshow(uint8(w)),title('V

                    R %

X,Y]=meshgrid(-OMEGA:OMEGA,-OMEGA:OMEGA);%%create G]

            ;(G=exp(-(X.^2+Y.^2)/2/sigma_s^2

            ;w1=double(w)/255

```

```

;(R=zeros(m,n

for i=1:m

for j=1:n

;(imin=max(i-OMEGA,1

;(imax=min(i+OMEGA,m

;(jmin=max(j-OMEGA,1

;jmax=min(j+OMEGA,n

;R1=R(imin:imax,jmin:jmax

;w11=W1(imin:imax,jmin:jmax

;(H=exp(-(w11-W1(i,j)).^2/2/sigma_r^2

;F = H.*G((imin:imax)-i+OMEGA+1,(jmin:jmax)-j+OMEGA+1

;((:)R(i,j) = sum(F(:).*w11(:))/sum(F

end

end

;figure,imshow(R,[ ]),title('R

vr %%

;Vr=zeros(m,n

for i=1:m

```

```

    for j=1:n

        ;(imin=max(i-OMEGA,1

        ;(imax=min(i+OMEGA,m

        ;(jmin=max(j-OMEGA,1

        ;(jmax=min(j+OMEGA,n

        ;(w11=w_double(imin:imax,jmin:jmax

        ;(R1=R(imin:imax,jmin:jmax

        ;(H1=exp(-(R1-R(i,j)).^2/2/sigma_r^2

        ;(F = H1.*G((imin:imax)-i+OMEGA+1,(jmin:jmax)-j+OMEGA+1

        ;((:)Vr(i,j) = sum(F(:).*w11(:))/sum(F

    end

    end

    ;('figure,imshow(Vr,[]),title('Vr

    T %

    ;([(A=min([Amin, max(max(255-W

    ;t=1-w/A*Vr

    ;('figure,imshow(t,[]),title('t

    J %

    ;(image_double=double(image

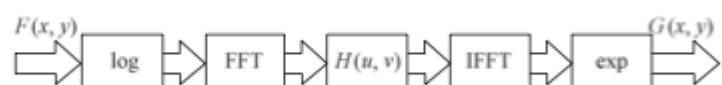
    ;J=(image_double-A)./max(t,t0)+A

```

```
;('figure,imshow(uint8(J)),title('J
```

## מאמר 2

האלגוריתם.



homomorphic filter formula

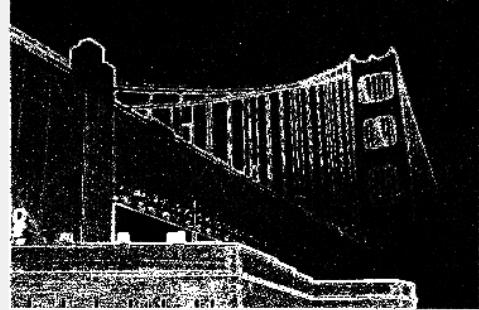
$$H(u, v) = (\gamma_H - \gamma_L) \left[ 1 - e^{-c} \frac{\left(\frac{H-u}{2}\right)^2 + \left(\frac{W-v}{2}\right)^2}{D_0^2} \right] + \gamma_L$$

בהתוצאה אפשר לראות שחלק מין האלמנטים אפר לראות רק לאחר הפילטר.

**org**



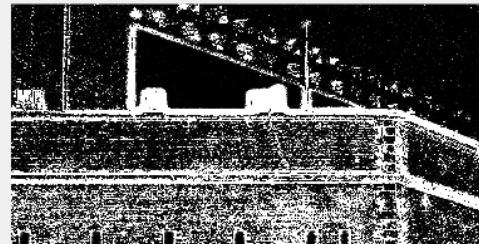
**new**



⠇ ⚡ ⌂ + 🔎 ⌂



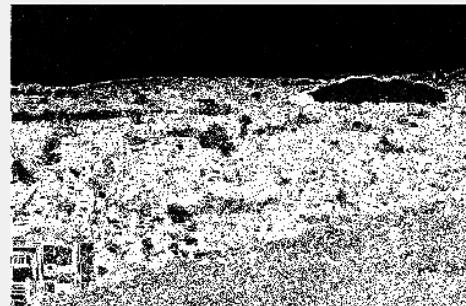
**new**



**org**



⠇ ⚡ ⌂ + 🔎 ⌂



תיקן