```
close all¿clear¿¿image1  = imread('gauss.jpg');¿image2  = imread('dyoung.jpg');¿¿N =
 length(image1);¿¿%% Create 5-point finite difference Laplacian on a square¿A = dels
q(numgrid('S',N+2));¿¿%% Increment the diagonal by diagonalIncrememement¿¿diagonalIncr
ememement = 0.5;¿A=A+diagonalIncrememement*speye(length(A));¿¿%% Create b such that imag
e2 is answer to Ax = b¿b = A*cast(reshape(image2,N^2,1),'double');¿¿%% Make image1 o
ur initial guess to Ax = b¿¿x0 = cast(reshape(image1,N^2,1),'double');¿¿%% Visualize
 initial guess¿¿image(reshape(x0,N,N))¿colormap(gray(256))¿axis equal off tight¿draw
now¿¿¿%% Set up matrix splitting of A¿D = diag(diag(A));¿L = -1*tril(A,-1);¿U = -1*t
riu(A,1);¿¿% The iteration matrix for Gauss-Seidel uses the lower triangular matrix
of A¿% instead of the diagonal.¿M = inv(D-L);¿¿%% Prepare diagonal splitting version
 of Jacobi method¿¿¿%% Perform Jacobi iteration¿¿x = x0;     % initial guess¿e0 = no
rm(x - cast(reshape(image2,N^2,1),'double'));    % initial error¿¿numIterations = 50
;     % number of iterations¿for iter = 2:numIterations¿¿    xold = x;           % u
pdate previous solution¿¿    x = M * (b + U * x);%M * (b + DSU*x);  % compute curren
t solution¿¿    % print out current error¿    eiter = norm(x - cast(reshape(image2,N
^2,1),'double'))/e0¿¿    % visualize current solution¿    cla¿    image(reshape(x,N,
N))¿    colormap(gray(256))¿    axis equal off tight¿    pause(.25)¿¿end¿
```