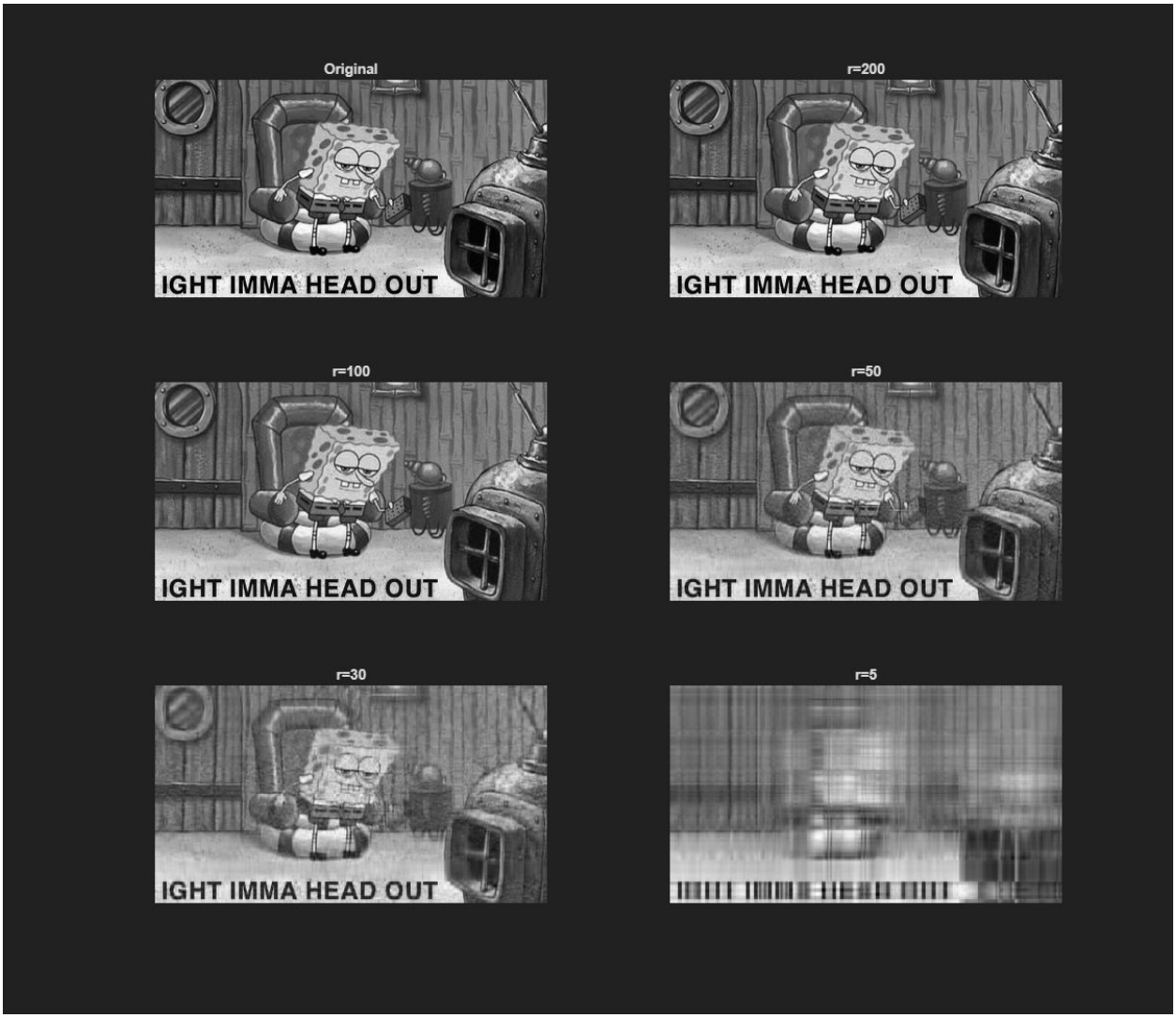
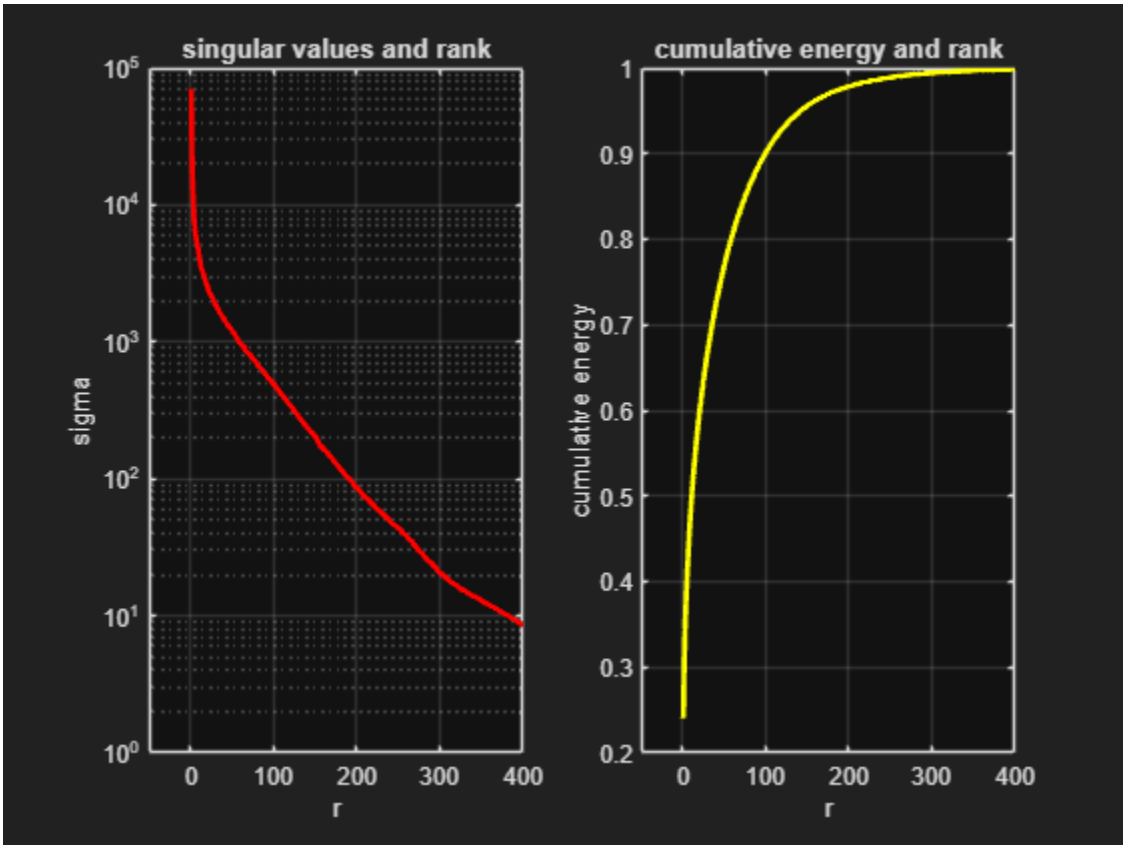
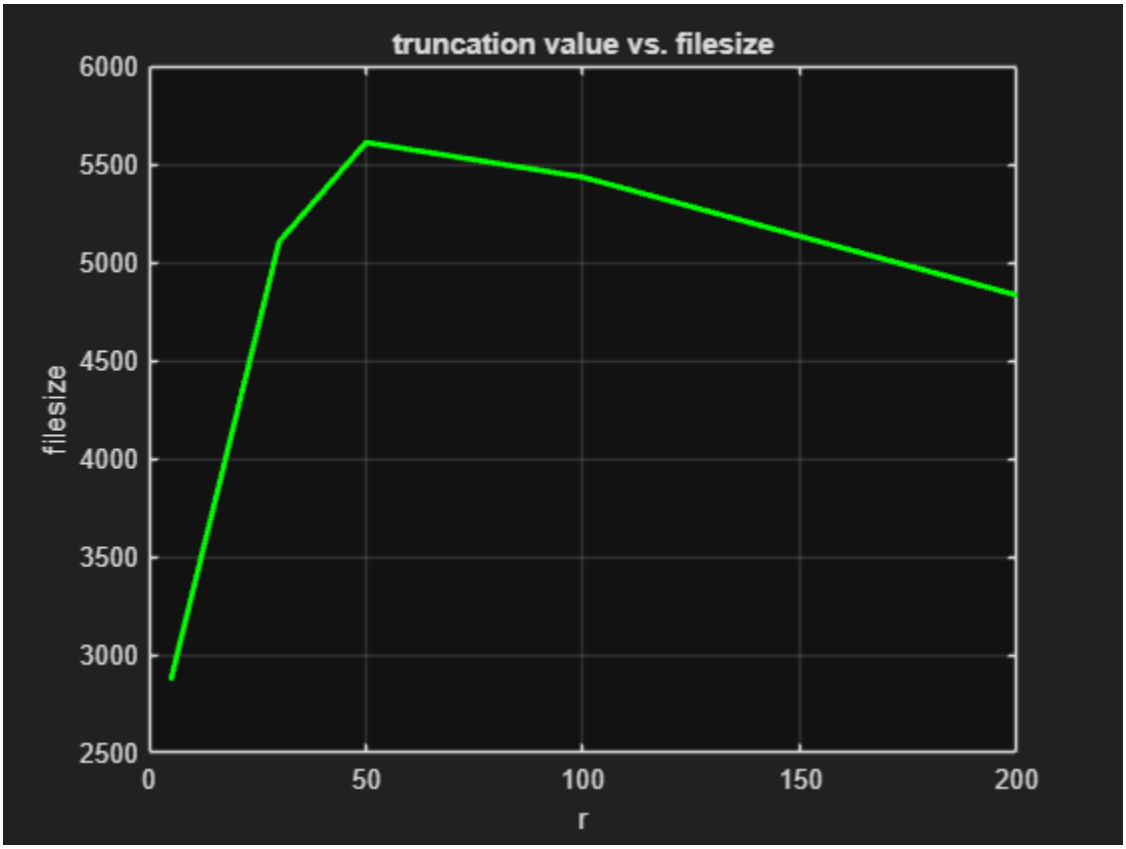
Table of Contents


```
clearvars; close all; clc

format='.png';

% read in an image and convert to grayscale
Image = imread('spongebob.png');
Matrix_Image = double(rgb2gray(Image));
imwrite(Matrix_Image, ['image_r=untouched', format])

% create plot
subplot(3,2,1)
imagesc(Matrix_Image), axis off
title('Original');
colormap gray
set(gcf, 'Position', [0 0 1200 1400])

[U,S,V] = svd(Matrix_Image, 'econ');

% range of truncation values
r = [200 100 50 30 5];
iminformation = zeros(1,length(r));

for index = 1:length(r)
    % approximate matrix using SVD
    Matrix_Approx = U(:, 1:r(index)) * S(1:r(index), 1:r(index)) * V(:,
1:r(index))';
    subplot(3, 2, index+1);
    imagesc(Matrix_Approx)
    imwrite(Matrix_Approx, ['image_r=', num2str(r(index)), format])
    iminformation(index) = imfinfo(['image_r=', num2str(r(index)),
format]).FileSize;
    axis off
    colormap gray
    title(['r=', num2str(r(index), '%d')])
end

% visualize singular value distribution
figure
subplot(1,2,1)
semilogy(diag(S), 'r', 'LineWidth', 2)
grid on
xlabel('r')
ylabel('sigma')
title('singular values and rank')
xlim([-50, 400])
subplot(1,2,2)
plot(cumsum(diag(S))/sum(diag(S)), 'y', 'LineWidth', 2)
```

```
grid on
xlim([-50, 400])
title('cumulative energy and rank')
xlabel('r')
ylabel('cumulative energy')


% determine effect of truncation on size
p = polyfit(r, iminformation, 4);
x = linspace(0, 1000, 1000);
y = polyval(p, x);
figure
plot(r, iminformation, 'g', 'LineWidth', 2)
xlabel('r')
ylabel('filesize')
grid on
title('truncation value vs. filesize')
```




singular values and rank
sigma
r
cumulative energy and rank
cumulative energy
r


truncation value vs. filesize
filesize
r

Published with MATLAB® R2023b