

Lab 1. Del 2- Laboration

Svarsdokument

Spara detta dokument som .pdf dokument innan ni lägger ut det på Lisam.

Studenternas namn och LiU-ID: (Max 2 studenter per grupp):

1. Emil Alsbjer, emial133

2. Saga Nyman, sagny722

Inlämningsdatum: 09/02 -24

Version(2)

Observera att ibland krävs det att ni ska skriva ett antal MATLAB kommandon som svar. Detta kan enkelt göras genom kopiera-klistra-in från MATLAB till svarsdokumentet.

Observera att ni kan skriva matriserna antingen genom att använda "Equation i words" och skapa matriser där, eller skriva dem precis som man definierar en matris i MATLAB, eller på ett annat lämpligt sätt för att tydligt visa en matris.

1) Variabeltyper och bildvisning

Uppgift 1.1)

Varför är $b/4$ inte helt korrekt?

Då b ska representera värdet 5 som en int, så kan inte decimaltalen visas. Därmed visas endast närmsta heltal. Dvs 1.

Vad får vi ut genom $b/12$? Och Varför?

$b/12$ visar 0. Detta av samma anledning som föregående fråga.

Uppgift 1.2)

Hur mycket mer minne behöver $k2$ jämfört med k och varför? (skriv hur du räknar)

$k2$ behöver ungefär 1.8MB mer minne.

$$\begin{array}{r} 2\,097\,152 - 262\,144 = 1\,835\,008 \\ 1\,835\,008 \\ \hline 1\,000\,000 \end{array} \sim 1.8$$

I k så tar varje pixel upp 1 byte minne. I $k2$ så är varje pixel en double vilket gör att varje pixel tar upp 8 bytes.

$$\frac{2\,097\,152}{262\,144} = 8$$

Uppgift 1.3)

Förklara varför medan bilden k visas som en korrekt bild, visas $k2$ som en helvit bild.

k ligger i rätt intervall $[0,1]$, medan $k2$ som är en double hamnar utanför intervallet och. Då varje pixel har ett värde högre än 1 så blir hela bilden vit.

Uppgift 1.4)

Förklara varför medan $\text{imshow}(k/255)$ visas som en helt svart bild, visas $\text{imshow}(k2/255)$ som en korrekt bild.

K2 blir normaliserad och ligger mellan intervallet [0,1] efter divisionen med 255. Därmed blir k2 en vanlig bild. K har nu väldigt små värden och representeras bara som helt svart

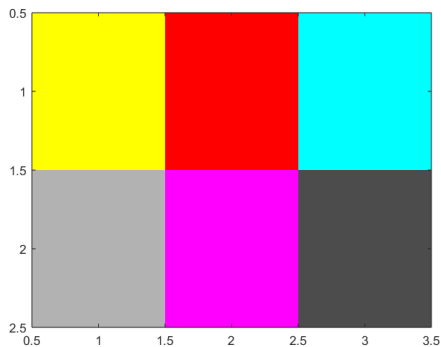
2) RGB färger och colormap

Uppgift 2.1)

Er MATLAB kod (**tre rader**) här: (ni kan kopiera och klistra in direkt från MATLAB)

```
map = ([1 1 0; 1 0 0; 0 1 1; 0.7 0.7 0.7; 1 0 1; 0.3 0.3 0.3]);  
image([1, 2, 3; 4, 5, 6]);  
colormap(map);
```

Infoga bilden här: (**Ok att skala ner bilden i words**)



3) Matriser och punktvis operation

Uppgift 3.1)

```
>> N(1:3:end, 1)= 1; 10  
>> N(1:3:end, :)= 1 2 3; 10 11 12
```

Uppgift 3.2)

Förklara anledningen om du får ett felmeddelande i någon av raderna.

s1=1

s2= Error.

Anledningen till felmeddelandet är v1 och v3 inte har samma storlek.

4) Logiska operationer

Uppgift 4.1)

u1= 0 0 0; 1 1 1; 1 1 1;

u2= 0 0 0; 0 0 1; 0 0 0;

u3= 0 0 0; 1 1 1; 1 1 1;

u4= 0 0 0; 0 0 0; 1 1 1;

5) Färgbilder

Uppgift 5.1)

MATLAB kommandon ni har använt för att skapa *mygray* här under (5-6 rader kod):

```
fargbild = imread('Butterfly.tif');  
fargbild = im2double(fargbild);  
imagesc(fargbild);  
mygray = (fargbild(:,:,1) + fargbild(:,:,2) + fargbild(:,:,3) /3);  
imwrite(mygray, 'mygray.png');
```

Infoga *mygray* här (skala inte bilden efter att ni har infogat):



6) Nedsampling och uppsampling

Uppgift 6.1)

MATLAB kommandot här: (går med bara en rad)

```
b61 = mygray(1:2:end, 1:2:end);
```

Infoga *b61* här (skala inte bilden efter att ni har klistrat in i words):



Uppgift 6.2)

Infoga *b62* här (skala inte bilden efter att ni har klistrat in i words):



Uppgift 6.3)

MATLAB kommandon här (det krävs bara en rad per bild):

```
b63_nearest = imresize(b61, 2, "nearest");  
b63_linear = imresize(b61, 2, "bilinear");  
b63_cubic = imresize(b61, 2, "bicubic");
```

Infoga *b63_nearest*: (skala inte bilden efter att ni har klistrat in i words)



Infoga *b63_linear*: (skala inte bilden efter att ni har klistrat in i words)



Infoga *b63_cubic*: (skala inte bilden efter att ni har klistrat in i words)



Diskutera vilken av dessa tre bilder och på vilket sätt ser bättre ut och liknar originalet mest: Nearest är väldigt pixlig medans cubic är väldigt oskarp. Linear har spår av både oskarpa och pixlar men inte till den grad som de övriga två. Därmed ser linear mest ut som originalet

Uppgift 6.4)

MATLAB kommandon här: (max två rader)

```
temp = imresize(mycolorimage, 0.5, "nearest");
b64 = imresize(temp, 2, "nearest");
```

Infoga *b64*:



Beskriv de tydliga skillnaderna mellan *mycolorimage* och *b64*:

B64 är betydligt mer pixlerad än mycolorimage

Uppgift 6.5)

MATLAB kommandon här: (max 10 rader men fullt möjligt med 6 rader)

```
R = mycolorimage(:, :, 1);
G = mycolorimage(:, :, 2);
B = mycolorimage(:, :, 3);
```

```
R2 = imresize(R, 0.5, "nearest");
R2 = imresize(R2, 2, "nearest");
```

```

B2 = imresize(B, 0.5, "nearest");
B2 = imresize(B2, 2, "nearest");
b65 = cat(3, R2, G, B2);
imwrite(b65, "b65.png");

```

Infoga *b65* här:



Vilken av bilderna *b64* och *b65* liknar originalet mest? Förklara varför!

B65 liknar originalet mer. Då grön färg innehåller mer information än röd och blå, så kommer den bild som inte har nedsamplat och uppsamplat den gröna färgen att likna originalet mer. Dvs manipulation av den gröna färgen har stor påverkan på bildens utseende.

Uppgift 6.6)

Hur mycket minne uttryckt i megabytes (MB) krävs för att spara en 4000×2000 pixlar stor **färgbild** i uint8-format? (skriv hur du räknat)

$$\begin{aligned}
 4\,000 * 2\,000 &= 8\,000\,000 \\
 8\,000\,000 * 8 * 3 &= 192\,000\,000 \\
 \frac{192\,000\,000}{8(1\,000\,000)} &= 24
 \end{aligned}$$

Svar: 24 MB

Hur mycket minne krävs för den komprimerade bilden om vi samplar ner två av bildens färgkanaler till hälften så stor i varje led men behåller den tredje som den var? (skriv hur du räknat)

$$\begin{aligned}
 4000 * 2000 &= 8\,000\,000 \\
 2(8\,000\,000 * 0.25 * 8) &= 32\,000\,000 \\
 8\,000\,000 * 1 * 8 &= 64\,000\,000 \\
 32\,000\,000 + 64\,000\,000 &= 96\,000\,000 \\
 \frac{96\,000\,000}{8(1\,000\,000)} &= 12
 \end{aligned}$$

Svar: 12 MB

Uppgift 6.7)

MATLAB kommandon här: (max 7 rader men fullt möjligt med 3 rader)

```

G2 = imresize(G, 0.5, "nearest");
G2 = imresize(G2, 2, "nearest");
b67 = cat(3, R2, G2, B);

```


Infoga [b67](#) här:



Förklara varför [b65](#) ser bättre ut än [b67](#) (är mer lik originalet).

Efter att ha nedsamplat och uppsamplat den gröna färgkanalen så har det större påverkan på bilden då grön färg innehåller mer information än röd och blå

Uppgift 6.8)

MATLAB kommandon här: (max 13 rader men fullt möjligt med 9 rader)

```
bild1 = R + G + B;  
bild2 = R - G;  
bild3 = R + G - 2*B;  
  
bild2 = imresize(bild2, 0.5, "nearest");  
bild2 = imresize(bild2, 2, "nearest");  
bild3 = imresize(bild3, 0.5, "nearest");  
bild3 = imresize(bild3, 2, "nearest");  
  
R3 = bild1/3 + bild2/2 + bild3/6;  
G3 = bild1/3 - bild2/2 + bild3/6;  
B3 = bild1/3 - bild3/3;
```

```
b68 = cat(3, R3, G3, B3);
```

Infoga [b68](#) här:



Ser ni att *b68* ser bättre ut än *b65* och är nästan identisk originalbilden?
Ja

Uppgift 6.9)

Hur mycket minne krävs för B1 (uttryckt i MB)? (skriv hur du räknat)

$$\begin{aligned}4\,000 * 12\,000 &= 48\,000\,000 \\48\,000\,000 * 3 * 8 &= 1\,152\,000\,000 \\ \frac{1\,152\,000\,000}{8(1\,000\,000)} &= 144\end{aligned}$$

Svar: 144MB

Hur mycket minne krävs för B2 (uttryckt i MB)? (skriv hur du räknat)

$$\begin{aligned}4000 * 12\,000 &= 48\,000\,000 \\48\,000\,000 * 3 * 8 * 0,0625 &= 72\,000\,000\end{aligned}$$

0,0625 då både höjd och bredd minskar med en faktor 0.025; $0.025 * 0.025 = 0.0625$

$$\frac{72\,000\,000}{8(1\,000\,000)} = 9$$

Svar: 9MB

Hur mycket minne krävs för B3 (uttryckt i MB)? (skriv hur du räknat)

$$\begin{aligned}4000 * 12\,000 &= 48\,000\,000 \\48\,000\,000 * 3 * 8 * 0,0625 &= 72\,000\,000\end{aligned}$$

0,0625 då både höjd och bredd minskar med en faktor 0.025; $0.025 * 0.025 = 0.0625$

$$\frac{72\,000\,000}{8(1\,000\,000)} = 9$$

Svar: 9MB

Glöm inte att spara dokumentet som *.pdf* innan ni lägger ut det på Lisam.