

# PM

## Modul Bilder

### Inledning

Jag tog itu med denna modul direkt efter att ha kunnat få upp min första WP-site enligt anvisningarna för modul 6. Så de första posterna var väldigt provisoriska under tiden jag försökte lära mig använda WP "inifrån" admin-panelen.

Jag läste först igenom den teoretiska delen "Bilder" på kursens hemsida och studerade länkarna som finns där. Det blev mycket nyttig information som gav svar på frågor jag länge hade undrat över:

Begrepp som ppi, dpi och slärmupplösning klarnade och under länkarna fann jag en del intressanta verktyg som jag började använda. T.ex. en digital linjal som man kan mäta pixlar beroende på vilken upplösning man har.

Jag mätte också upplösningen på mina egna skärmar och fann att de låg på värden mellan 91 – 112 ppi.

Ett viktigt begrepp som klarnade helt är "dimensionen" av en pixel och hur den kan vara relativ. Hittills hade jag bara lyckats få en ungefärlig förståelse av detta, men nu kunde jag handgripligen se det genom att ställa in skärmarna i olika upplösningar, hur pixlarna förstorades och förminskades beroende på skärmupplösning.

På ett ganska tidigt stadium började jag också misstänka att min Photoshop inte skulle räcka till för att lösa alla uppgifter. Jag gjorde då det som jag länge velat men aldrig skridit till verket med, att ladda ned och installera GIMP. Det var en positivt laddad erfarenhet att lära mig och börja arbeta med det nya gränssnittet. Jag såg snart att man hade fler möjligheter för komprimering med GIMP än vad jag kände till genom PS.

## Del 1 - Komprimering

En litet förvirring jag först stötte på (och som senare togs upp av lärare i Forumet under Modul 2), var begreppen för filstorlek vad gäller datalagring vs dataöverföring. Alltså att 1kb = 1024 b, medan 1kpbs = 1000 bs. Fick känslan att begreppen ofta blandades ihop på nätet och undrade hur lösa uppgifterna om det råder sådan förvirring. Men ju mer jag arbetade med uppgifterna desto mer klarnade dessa begrepp.

(En annan senare förvirring var den som jag också tog upp i forumet om färgdjupet: När jag räknade på vissa bilder enligt formlerna så fick jag värden mindre än 24 bpp, som ju skulle vara vad som krävdes för en bild i full färg. Efter diskussionen i forumet så blev kom jag fram till att när man anger kompressionsgraden i bpp inte betyder att antalet färger anges med färre bitar utan att det med den använda kompressionmetoden använder i medel mindre bitar per pixel för att representera bilden.)

### PNG – Förlustfri komprimering

PNG använder sig av en pixels närliggande pixlar för att med en algoritm förutsäga dess färg, samt zip-komprimerar felet av förutsägelsen.

Det var då dags att praktiskt börja med de första komprimeringarna. Jag arbetade med png komprimering i GIMP eftersom där som sagt fanns utökade val att göra.

Formeln för komprimeringen är:

$$R = S/M \times N$$

Där R = kompressionsgraden

S = filens storlek i bits

M, N bilden dimensioner i pixlar.

I Gimp kan man ställa in 0-9 olika nivåer för png-komprimering.

Nivå 0 ger ingen komprimering alls, kompressionsgraden blir 1:1 och resultatet blir 100% av originalbilden.

Nivå 9 (den hårdaste komprimeringen) ger genom uträkning följande kompressionsgrad: 14bpp = 60,5% = 1,6:1.

Detta är c:a 5% mer än vad man kan åstadkomma med Photoshop.

Jag provade också att konvertera tillbaka till tiff-formatet och fick då en bild som var precis lika stor som originalbilden, 768 kB.

### JPEG – Förlustintroducerande komprimering

JPEG använder sig av en helt annan metod för komprimering vilken gör att man förlorar data som inte kan återskapas.

Enligt formeln ovan får man följande storlek på jpeg-bilderna med angivna bpp:s:

- 4 bpp -> 128 kB
- 2 bpp -> 64 kB
- 1 bpp -> 32 kB
- 0,5 -> 16 KB
- 0,25 bpp -> 8 kB

Den uträknade kompressionsgraden (png's bpp) var 14,5 bpp och borde ge en storlek 464 kB.

Jag fick max 405 kB när jag försökte . För att åstadkomma detta använde jag GIMP's advanced options:

- Optimize
- Progressive
- Subsampling: best quality
- DCT method: fast Integer

Det är de olika parametrarna man kan variera för att ändra resultatet. Lägre komprimering är så (bättre kvalité) kan inte GIMP ge.

## **Del 2 - CSS-Sprites**

Jag hade förut stött på CSS-Sprites men aldrig använt det praktiskt. Men efter att ha följt tutorials på nätet och efter viss krångel med precisionen, lyckades jag få till en CSS-sprite såsom den presenteras på sajten under Images. Jag följde dock inte den T-liknande strukturen utan lade bilderna i följd efter varandra. Jag hade frågat om detta i forumet och fått ok av lärare. Man ser där tydligt att kvalitetsförsämringen som JPEG introducerar blir uppenbar vid kompressionsgrad 2bpp och lägre .