

**=== HOW TO USE + EXTRA TIPS**

---

\*

\* **Radius is the radius as known from standard USM**

\* **Because here a fast infinite impulse response-filter is used, very high radius**

**is possible without slowdown.**

- \* **This comes in handy, when USM is used to enhance local contrast.**
- \* **I do this often, and this was very very slow with gimp's original USM.**

\*

- \* **I use a sony dsc f717 that is known for "flat" images (low contrast) and also**

**for high dynamic**

- \* **range. Using USM, rather than curves, I can enhance contrast without**

**losing shadows or highlights,**

- \* **its just contrary: shadows and highlights get more (local) contrast, and no**

**detail is lost.**

- \* **(For real strong fill-flash effect a contrast mask must be used in gimp and**

**afterwards local contrast**

- \* **can be enhanced using USM)**

\* .....

\*

- \* **Amount+ and Amount- are the amount factors, as known from USM,**

**however, "+" means amount of**

- \* **brightening and "-" means amount of darkening**

- \* **Threshold is the treshold as known from gimps USM**

\* .....

\*

- \* **Gamma changes the gamma factor for internal processing. High gamma**

**values can reduce sharpening halos.**

- \* **The gamma of the resulting image is not changed.**

- \* **(Try gamma of 5.0 and +/- amount upto 5.0. This sounds unreasonable, but**

**it works very good )**

\*

- \* **Why?**

\*

- \* **Most digital cameras have a gamma value from 1.5 to 2.0.**

- \* **When the image is in computer memory then the function between**

**intensity i in nature and intensity**

- \* **value m in computer memory is :  $m = \text{pow}(i, 1/g)$ . (Where g ranges from 1.5**

**to 2.0)**

- \* **Thats similar to a squareroot-curve.**

- \* **When we apply USM to this unmodified image, then we get more bright**

**halo than dark halo.**

\* This looks very un-natural.

\* When we linearize data before doing USM, then we get equal amounts for

dark and bright halos.

\* When we overcompensate g then we get dark halos, and in most cases

these are unvisible or

\* look natural.

\*

\* .....

\*

\* Highlight-,Light-,Midtone-,Shadow Amount

\*

\* This can be used, to constrain the filter amount to darker or lighter areas of

the image.

\* Especially when USM is used to enhance local contrast it is useful.

\* For normal sharpening these sliders can be left at 1.0

\*

\* Preserve currently does nothing in sharpening mode.

\* In a later version this will protect color, that means it will prevent color

artifacts

\*

\* ..... Special feature, Blur and denoise

.....

\*

\* When both, Amout+ and Amount- are set to zero, then USM2 goes into

blur mode.

\* The image is blurred with given radius.

\* The blurred image is copied into the Output image, where the given

Threshold is used as

\* criteria. As long Threshold is 0.0 no blur happens. When Threshold is

larger, then we get an effect

\* similar to selective gaussian blur. When Threshold is 1.0, then we get pure

gaussian blur.

\* Gamma value can improve denoising results, it works with best selectivity

at gamma values  $\leq 0.5$

\* Highlight Amount ... Shadow Amount can be used to constrain the

filtereffect to brightness ranges.

\*

**\* Because the same threshold can be used for sharpening and for denoising,**

**this comes in handy when**

**\* USM has amplified some noise in sharpening mode, or when denoised**

**pictures should be sharpened.**

**\***

**\* Preserve protects original luminance while denoising/ blurring.**

**\* This comes in handy when we want to remove color noise in shadows**

**before or after sharpening.**

**\***

**\* I have also written another denoising tool (dcamnoise). However,**

**sometimes, when I see the results**

**\* which I get with this modified USM version here, then I am astonished what**

**it can do, when the parameters are**

**\* carefully adjusted. Especially with low frequency noise (that is already**

**filtered by camera firmware) it**

**\* is often better.**

**\***

**Credit**

**this plugin was created by Peter Heckert , the windows binary by courtesy of**

**Michael Schumacher.**

**Source Code is included**