

040b73747265616d747970656481a203840163c48403737373810a0a810b
0b815f5f84012584067f411b312d37OneVision-Image: Bildkanaloperationen
TMSImgChannel.tiff ↗ Bildkanaloperationen

Dieses Werkzeug ist eine Erweiterung zum Werkzeug *Bildkanäle* und entspricht in etwa dem Modus *Import* (;TMSImgChannel.rtf;Import;↗) dieses Werkzeugs. Das aktuelle Bild muß mit einem anderen Bild verknüpft werden.

Während aber *Import* einen Kanal des verknüpften Bildes in beliebige Kanäle des aktuellen Bildes kopiert, können hier die Kanäle über Funktionen verknüpft werden. Es stehen zunächst folgende Grundoperationen zur Verfügung:

paste.tiff ↗

Abb.: Vordefinierte Formeln in der Funktion Bildkanaloperationen.

Begriffe

q steht hierbei für den Kanal der Quelle und
z für den Zielkanal.

Bedingungen

LT (a , b): **Less Than** ± eine Vergleichsfunktion, die den Wert 0 oder den Wert 1 als Ergebnis liefert. Ist $a < b$ so ist das Ergebnis 1 sonst 0.

GT (a , b): **Greater Than** ± eine Vergleichsfunktion, die den Wert 0 oder den Wert 1 als Ergebnis liefert. Ist $a > b$ so ist das Ergebnis 1 sonst 0.

EQ (a , b): **EQual** ± eine Vergleichsfunktion, die den Wert 0 oder den Wert 1 als Ergebnis liefert. Ist $a = b$ so ist das Ergebnis 1 sonst 0.

IF(a,b,c): Das Ergebnis dieser Funktion ist entweder b oder c. Ist a von 0 verschieden, so ist das Ergebnis b , sonst ist es c.

ABS (q - z)

Bildet die Differenz der Werte in den beiden Kanälen, und nimmt davon den absoluten Wert. Der resultierende Wert ist somit der Unterschied der beiden Kanäle. Diese Formel ist vor allem interessant um zwei Bilder zu vergleichen (z. B. vor und nach einem Filter).

(q + z) * 0.5

Mischt die beiden Kanäle im Verhältnis 50% zu 50 %.

AND (q, z)

logische UND-Verknüpfung (Interessant für Bitmaps).

OR (q, z)

logische ODER-Verknüpfung (Interessant für Bitmaps).

XOR (q, z)

logische EXKLUSIV-ODER-Verknüpfung (Interessant für Bitmaps).

q

Die Quelle wird einfach kopiert (entspricht somit dem *Import*).

NOT (z)

Das Ziel wird invertiert (Die Quelle schaut nur dumm weil sie nicht beteiligt wird).

NOT (q)

Das inverse der Quellkanals wird in die Zielkanäle übernommen.

Die nächsten fünf Operationen benutzen folgende Funktionen:

IF (e,a,b) Vergleich, Wenn der Ausdruck e wahr ist

dann a ansonsten b.

LT(a,b)	Lower Than ,	Ist wahr wenn $a < b$
GT	Greater Than ,	Ist wahr wenn $a > b$
EQ	Equal ,	Ist wahr wenn $a = b$

IF (LT (q , z) , q , z)

Setzt den kleineren Wert der beiden Kanäle. Die Konstruktion entspricht also:

`if(q<z) { z=q; } else { z=z; }`

IF (GT (q , z) , q , z)

Setzt den größeren Wert der beiden Kanäle.

IF (LT (q , z) , 0 , 1)

Setzt den Wert 0 (bei Graubild Schwarz) falls $q < z$ sonst 1 (bei Graubild Weiß).

IF (GT (q , z) , 0 , 1)

Setzt den Wert 0 (bei Graubild Schwarz) falls $q > z$ sonst 1 (bei Graubild Weiß).

IF (EQ (q , z) , 0 , 1)

Setzt den Wert 0 (bei Graubild Schwarz) falls $q = z$ sonst 1 (bei Graubild Weiß). Zum Vergleichen zweier Kanäle. Handelt es sich z. B. um zwei Graubilder, so wird das aktuelle Bild Schwarz, außer an den Stellen, wo sich die beiden Bilder unterscheiden. Dort wird das Bild weiß.

Mit dem Befehl `<Neu¼>` erhält man folgendes Fenster in das man selbst eine Funktion eingeben kann:

315584_paste.tiff ↵

Abb.: Das Eingabefenster für eigendefinierte Formeln für Kanaloperationen.

Nach dem Eingeben erscheint die neue Funktion im Pop-Up-Menü, falls der Ausdruck göltig war. Die Werte der Kanäle bewegen sich hier im Bereich von 0 bis 1 (d.h. der Ausdruck "1-z" invertiert das Ziel).

Wird eine selbstdefinierte Funktion benutzt, so ist beim ersten Aufruf einige Vorarbeit nötig, die je nach Rechnertyp und Formel unterschiedlich lang dauert. In dieser Zeit verwandelt sich der Schalter <neu> in eine Prozentanzeige.

Angaben zum Formelinterpreter

Eingebaute Operationen und deren Rangfolge:

~	2-er Komplement (long)
^	Potenzierung
* /	Multiplikation, Division
+ -	Addition, Subtraktion
<< >>	Shiften nach links (=*2), Shiften nach rechts
(=/2)	
&	bitweise UND-Verknöpfung (long)
	bitweise ODER-Verknöpfung (long)

Erlaubte Zahlenformate:

Binär:	%1111,%00101
Hex:	\$0f,0xfF0,0X0FF
Okt:	0127,0888
Dez:	1232,-12,4E-17,-4E3

Eingebaute Konstanten sind pi und e (muß klein geschrieben werden), z.B. 2pi wird als 2*pi ausgewertet.

Sollte der Buchstabe 'X' als Variable verwendet werden, so ist 0x für 0*x nicht erlaubt

Zwischen Klein- und Großschrift wird nicht unterschieden außer:

- e -> bedeutet die Eulerische Konstante
- E -> als Exponent einer Dezimalzahl

Eckige Klammern sind nicht erlaubt.

ACOS(a)
ACOSH(a)
ASIN(a)
ASINH(a)
ATAN(a)
ATANH(a)
CEIL(a)
COS(a)
COSH(a)
ABS(a)
FLOOR(a)
LOG(a)
SIN(a)
SINH(a)
SQRT(a)
TAN(a)
TANH(a)
LB(a)
LG(a)
SQR(a)
FACT(a)
MIN(a,b)
MAX(a,b)
COT(a)
ACOT(a)

Bitte beachten: Die neu erstellten Funktionen sind temporär, d. h. sie können weder geladen noch gespeichert werden. Sie sind nach dem Programmende verloren.