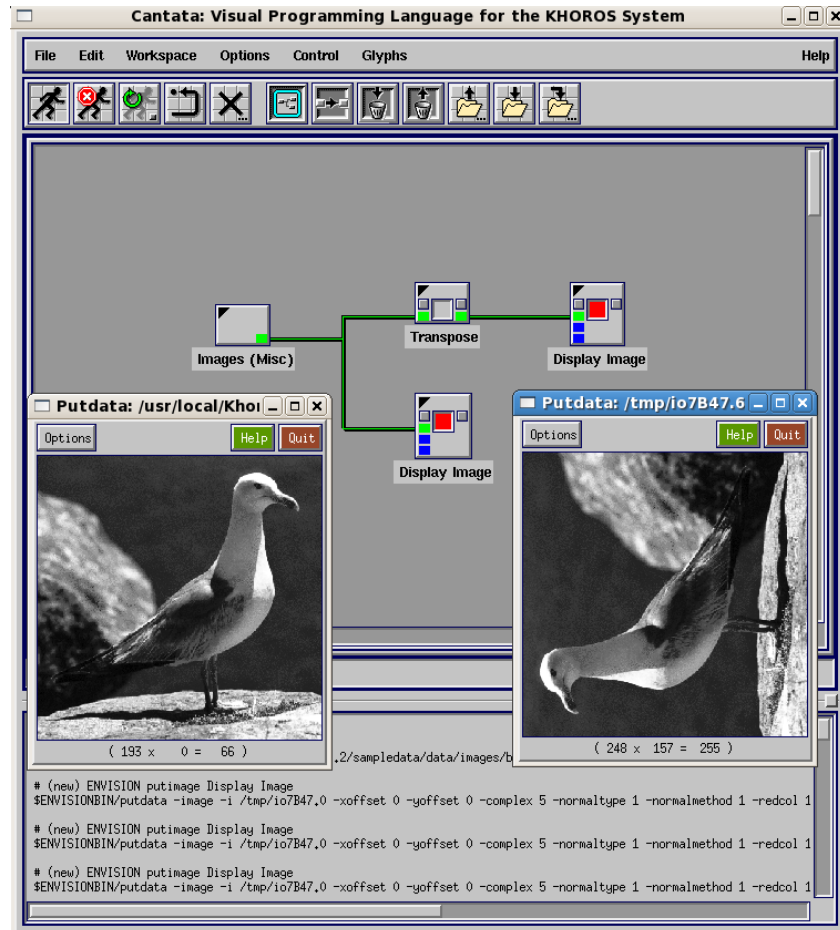


2. Gyakorlat

Khoros Cantata

Ismerkedés a Khoros Cantata-tel:

A Khoros Cantata egy képfeldolgozó műveletsorok készítésére szolgáló program. A műveleteket csővezetékszerűen lehet egymás után kötni. A műveleteket dobozokkal jelölik, melyeknek lehetnek bemenetei és kimenetei. A bemenetek mindig a doboz bal oldalán, a kimenetek a doboz jobb oldalán találhatók.



A Khoros Cantata munkaablaka (Linux)

Az ablak tején az Eszköztár található. Az Eszköztáron található gombok balról jobbra:



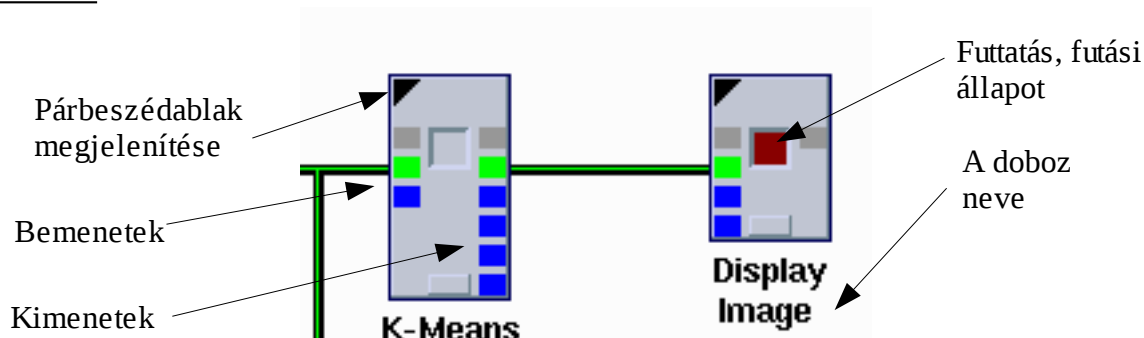
Khoros Cantata eszköztár (Linux)

- Műveletsor futásának indítása
- Műveletsor futásának megállítása
- Futási mód (változások végrehajtása után azonnali többszörös futtatás, vagy egyszeres futtatás)
- Alaphelyzetbe állítás
- Munkaterület törlése
- Egy komplett eljárás beillesztése

- Kiválasztott doboz klónozása
- Kiválasztott doboz kukába dobása (törlése)
- Kidobott doboz visszavétele
- Workspace file megnyitása
- Workspace file mentése
- Workspace mentése új fájlként

A műveleteket a Khoros Cantata *Glyph*-eknek hívja. A *Glyph*-eket a **Glyphs** menüből választhatjuk ki vagy rá is kereshetünk. Az egyes műveletekre most nem térek ki, a gyakorlatok során mindig lesz írva, hogy a feladathoz szükséges művelet hol található.

Dobozok:



A dobozok bemeneteit/kimeneteit különböző színekkel jelöli a program:

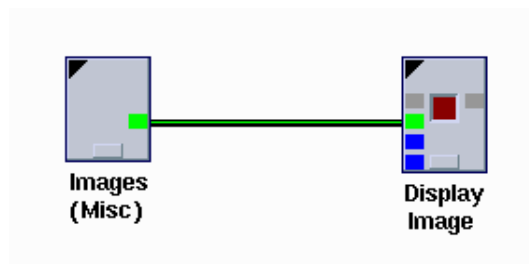
- zöld: érvényes adat
- sárga: szükséges adat
- kék: opcionális bemenet/kimenet

A dobozok bal felső sarkában lévő kis háromszögre kattintva elérhető a doboz párbeszédpanelje. A doboz közepén lévő négyzetre kattintva művelet elindítható, illetve látható, hogy fut-e.

Első műveletsor: kép megjelenítése

Válasszunk egy Khoros Cantata-ban lévő képet és jelenítsük meg.

1. Rakjuk a munkaterületre a **Glyphs** → **Input/Output** → **Data Files** → **Image(Misc)** dobozt. Windows-os verziónál ezt ki tudjuk keresni a Glyph-keresőben.
2. Rajuk a munkaterületre a **Glyphs** → **Input/Output** → **Visualization** → **Non-Interactive Image display** → **Display Image** dobozt. A Windows-os verzióban ezt is meg lehet találni a keresőben.
3. Kössük össze az **Image(Misc)** doboz zöld kimenetét a **Display Image** sárga bemenetével.
4. Válasszunk egy képet az Image (Misc) párbeszédpaneljén.
5. Indítsuk el a műveletsort.



Képmanipuláció

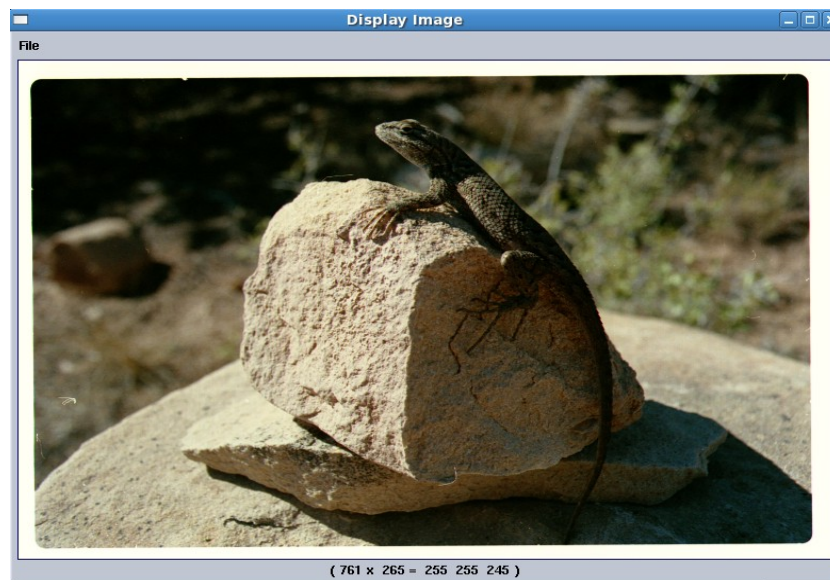
Az egyszerű képmanipulációk közé olyan műveletek tartoznak, mint például az eltolás, képrészlet kivágása, átméretezés, geometriai transzformációk (forgatás, skálázás, eltolás), tükrözés, ...

Képek, adatok

A legtöbb képfeldolgozó program (talán lehet néhány kivétel) a kép origóját a bal felső sarokba rakja, és a következő koordináta-rendszert használja:



A kép két fontos jellemzője a szélesség és a magasság, ezen kívül fontos az is, hogy milyen adattípust használnak a képelemek tárolására. A pixelek lehetnek két elemű (fekete és fehér), egészek, valós számok és komplex számok is. Valós számok esetében meg vannak engedve a negatív értékek is. Az ábrázolási tartomány a $[-1.0, 1.0]$ intervallum. Komplex számokkal a diszkrét Fourier-transzformációnál találkozhatunk. Színes képek esetében általában az RGB színmodellt használják. Ilyen színes képeket találhatunk a **Glyphs** → **Input/Output** → **Data Files** → **RGB Images** dobozban.

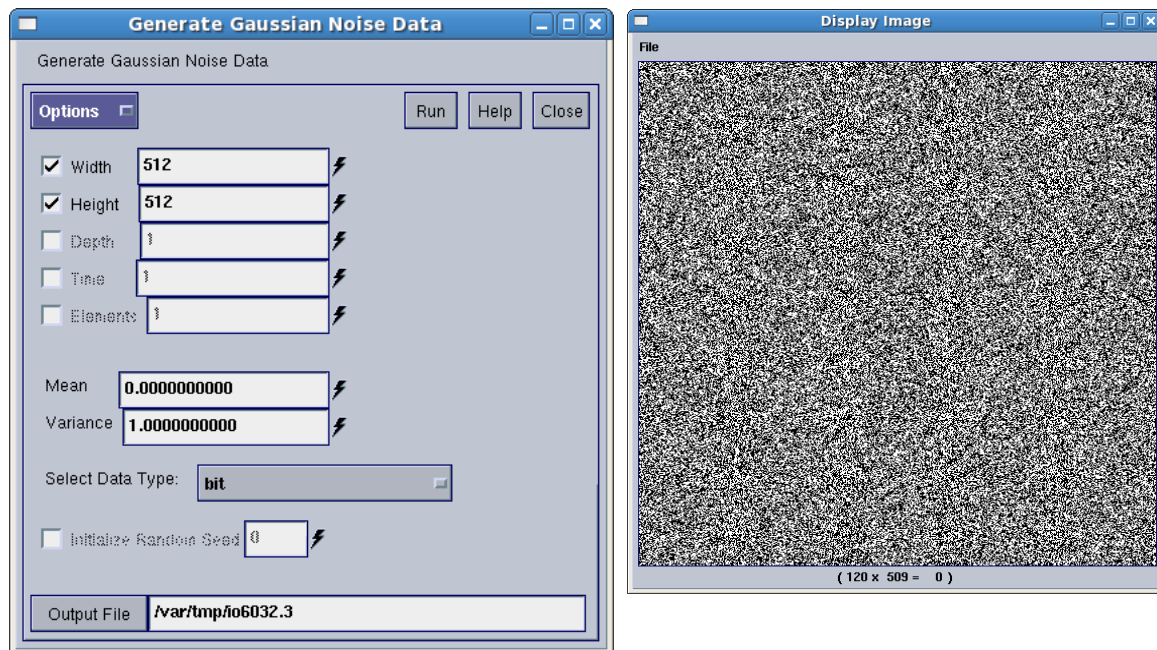


Színes kép az RGB Images dobozból

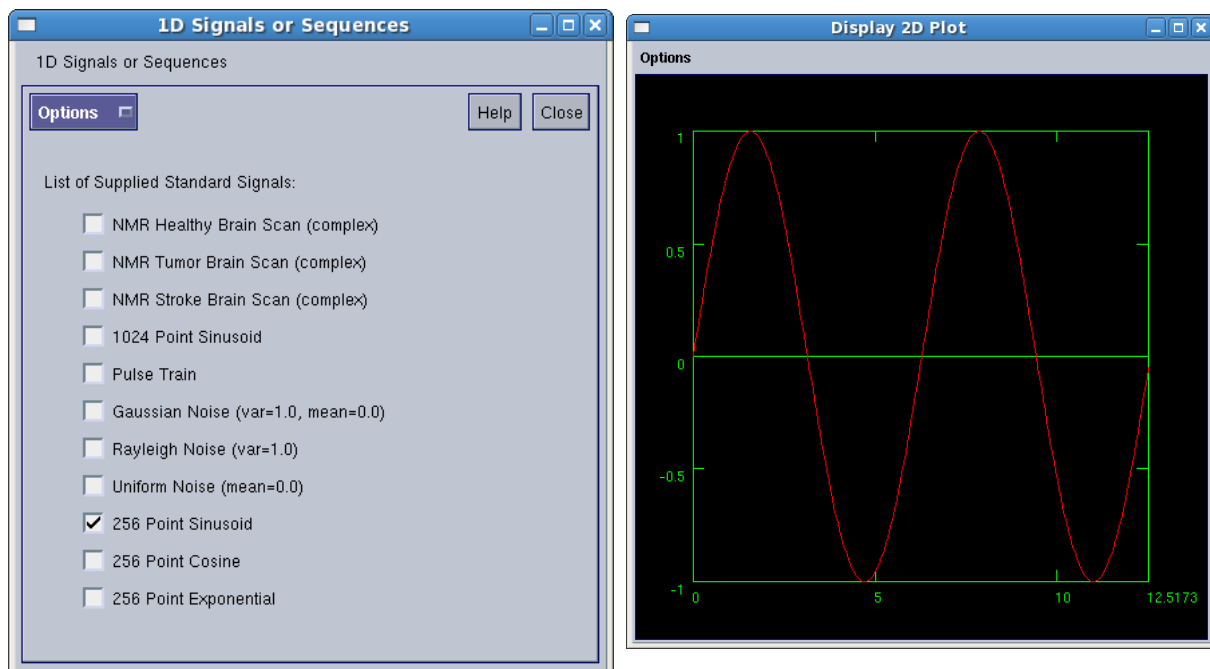
Nem csak a beépített képeket lehet használni, hanem saját képeket is megnyithatunk a programban. Ehhez használjuk a **Glyphs** → **Input/Output** → **Data Files** → **User defined** dobozt. Ebben a dobozban ki lehet választani a saját képünket, és fel lehet használni a műveletsorban.

A képeken kívül input adatként használhatunk egy konstans értéket generáló dobozt, vagy olyat amely valamilyen függvényt vagy zajt generál. Ezekre akkor van szükség, ha pl. valamilyen képjavító eljárást fejlesztünk és szeretnénk tesztelni zajos képen. A Khoros Cantata-ban több zajt is be lehet generálni, így több típusú zaj esetében lehet vizsgálni pl. egy simító maszkot. Az egyes zajtípusokat generáló dobozokat a **Glyphs** → **Input/Output** → **Generate Noise** menüpont alatt

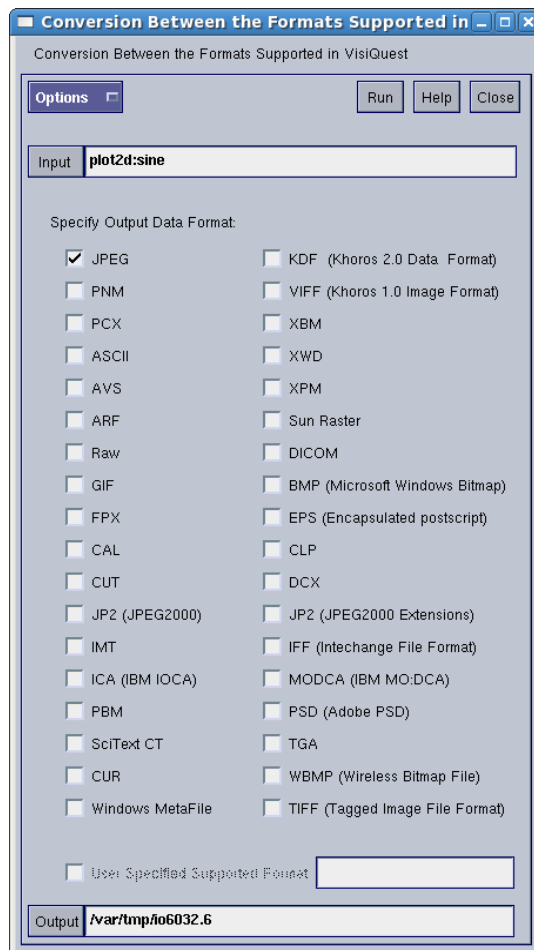
találjuk. A generált zaj is tulajdonképpen egy képként áll elő. Az alábbi ábrán egy bináris Gauss-zaj látható. Természetesen a zaj adattípusát és a kép méretét is beállíthatjuk.



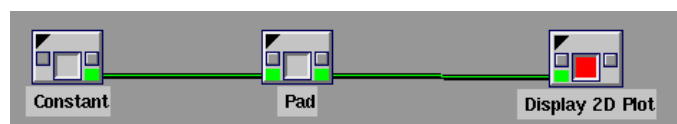
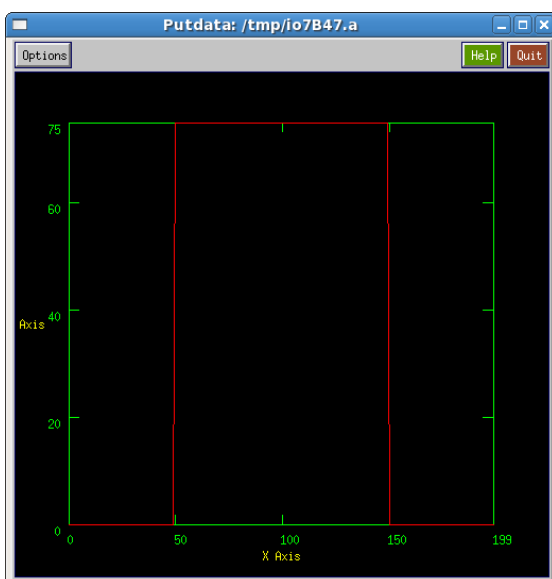
Lehetőség van függvények generálására is. Ehhez használjuk a **Glyphs → Input/Output → Data Files → Signals (1D)** dobozt. Ezek a függvények megjeleníthetők képként, de megjeleníthetjük őket függvényként is. Ha képként jelenítjük meg őket, akkor csak egy nagyon vékony képet látunk, és ezen a függvényt, mint intenzitásváltozást láthatjuk. Ha magát a függvény menetét szeretnénk megnézni, akkor használjuk a **Glyphs → Visualization → Plot Display → Display 2D plot** dobozt.



Egyéb megjelenítők, függvényrajzolók a **Glyphs → Visualization** menüpont alatt találhatók. Az egyes eredmény képek természetesen le is menthetők a **Glyphs → Input/Output → Export Data → Supported formats** dobozzal.



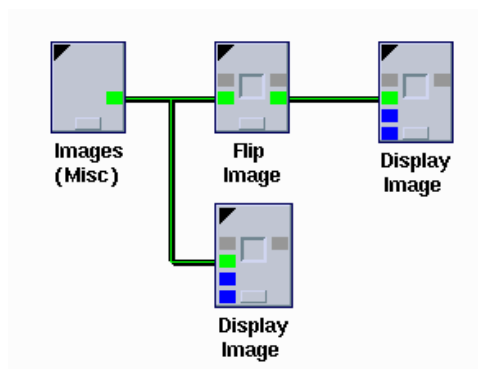
Konstans érték definiálására is van lehetőség. Nézzük meg, hogy az alábbi eredményt hogyan tudjuk elérni. Válasszuk a **Glyphs** → **Input/Output** → **Generate Data** → **Constant** dobozt. A **Glyphs** → **Data Manip** → **Size & Region Operators** → **Pad** dobozzal meg tudjuk növelni a kép méretét. Most csak a szélesség dimenziót használjuk.



Egyszerű képmanipulációk

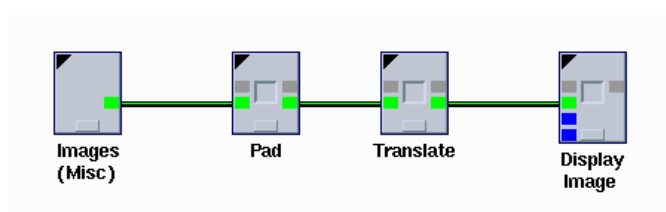
Tükrözés

Az első képmanipuláció legyen az, hogy a sírályos képet tengelyesen tükrözzük. Ehhez használjuk a **Glyphs → Image proc → Geometric Operators → Flip Image** dobozt. Itt beállíthatjuk, hogy a képet mely tengely mentén szeretnénk tükrözni.



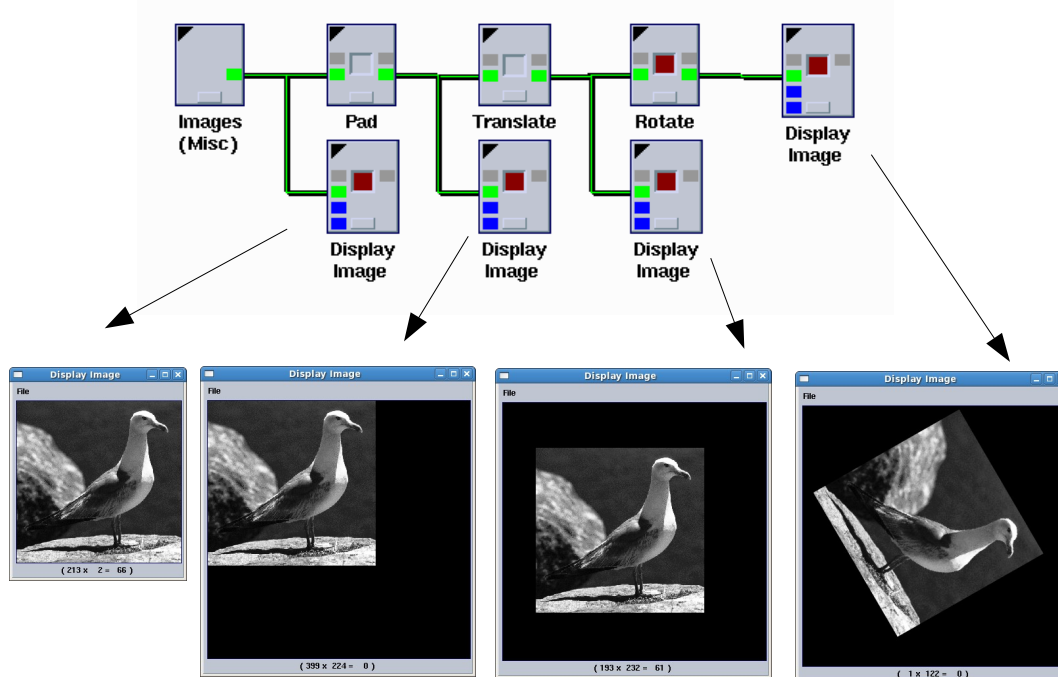
Eltolás:

Toljuk el az input képet egy kicsit jobbra és le. Használjuk a **Glyphs → Data Manip → Reorganize Data → Translate** dobozt. A kép egy része el fog tűnni, mivel a rajzterület eredetileg akkora, mint a kép. A rajzterület méretét meg tudjuk növelni a **Glyphs → Data Manip → Size & Region Operators → Pad** dobozzal. A rajzterület azon részén, ahol nincs kép, konstans érték fognak szerepelni.



Forgatás:

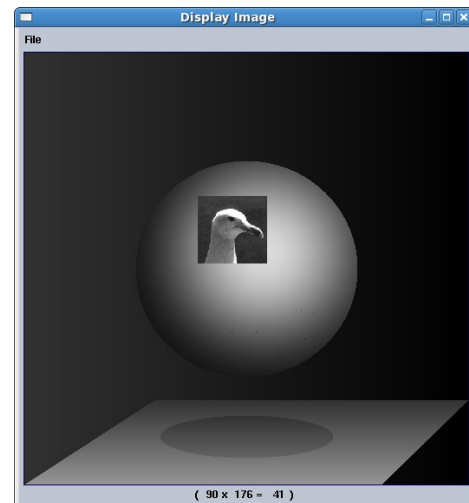
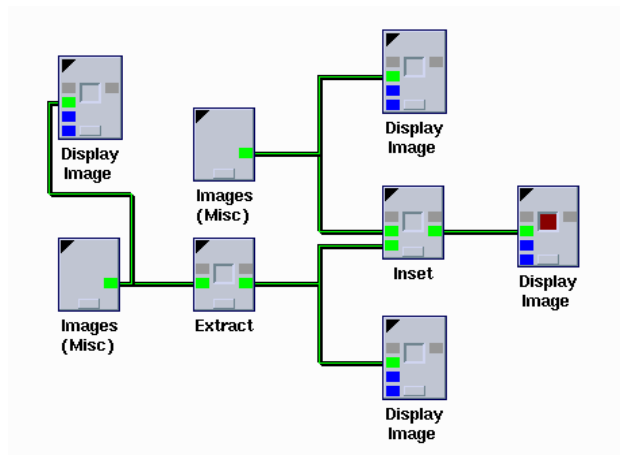
A forgatáshoz a **Glyphs → Image Proc → Geometric Operators → Rotate** dobozt kell használni. Azonban érdemes arra figyelni, hogy a forgatás során a kép néhány része le fog kerülni a rajzterületről. Ezért érdemes megnövelni a rajzterületet a **Glyphs → Data Manip → Size & Region Operators → Pad** dobozzal, eltolni a képet a rajzterület közepére és utána végrehajtani a forgatást.



Képrészlet kivágása:

A kép egy általunk kiválasztott téglalap alakú területét a **Glyphs** → **Size & Region Operators** → **Extract** dobozzal vághatjuk ki. Itt megadhatjuk hogy mekkora legyen a téglalap területe és mi legyen a kivágás origója (a téglalap bal felső sarka).

Most ezt a képrészletet másoljuk be egy másik képbe. Ehhez a **Glyphs** → **Size & Region Operators** → **Inset** dobozt használjuk.



Feladat:

Állítsd elő az alábbi képeket! A fehér téglalap méretei 15x40, a képméret 256x256. A forgatás szöge 45°. A fehér téglalap bal felső sarka a kép közepén legyen!

