

Felhasználói dokumentáció

A program segítségével PPM formátumú képeket lehet módosítani. A program parancssorral vezérelhető. A program a parancssorban megadott kapcsolókhoz tartozó műveleteket a felsorolás sorrendjében végzi el.

-h, --help	ezen menü megjelenítése és kilépés
-i, --input fájl	bemenetként használt képfájl útvonala
-o, --output fájl	kimeneti kép útvonala
--lightness ±érték	a kép fényességének változtatása, +fényesebb, -sötétebb
--contrast ±érték	a kép kontrasztjának állítása, +nagyobb kontraszt, -kisebb kontraszt
--hue-shift ±érték	a kép HSL hue értékének eltolása a megadott értékkel
--invert	a kép negatívvá tétele
--sinecolor-shift frekvencia	a kép színeit a szinusz függvény alapján torzítja
--mirror típus	a kép tükrözése, típus: diagonal, horizontal, vertical irányokban
--rgb-shift ± rvalue, ± gvalue, ± bvalue	RGB shift alkalmazása a képen, a színek értékeit a megadott értékekkel csúsztatja el a megfelelő irányba
--pixelsort preset	pixelsort algoritmus végrehajtása a képen a megadott preset alapján preset: edges: megkeresi a kép objektumainak a szélét és ezek között rendez all-random: teljesen véletlenszerű beállítások landscape: tájképekhez és nagy tárgyakhoz macro: részletes képekhez használható fewcolors: kevés színt tartalmazó képekhez dark: sötét területek kiemelése
--blur érték	a kép elmosása a megadott értékkel arányosan, kis érték kis elmosás, nagy érték nagy elmosás
--sharpen érték	a kép élesebbé tétele a megadott értékkel arányosan, kis érték kis élesítés, nagy érték nagy élesítés
--corrupt	teljesen véletlenszerűen tönkreteszi a képet
--grayscale	a kép fekete-fehérre változtatása
--3d	a képet vörös-cián 3D képpé alakítja
--edge-detect	a kép objektumainak függőleges széleit mutató képet adja vissza

Funkciók leírása és példák

A példák az alábbi input.ppm képet használják:



A képet [@ballibbelkeltem](#) készítette.

lightness

A kép fényességét állítja. Ajánlott értékei: -100 és +100 között.

```
./imageproc -i input.ppm -o output.ppm --lightness 22
```



contrast

A kép kontrasztját állítja. Ajánlott értékei: -80 és +260 között. A 259 nem megengedett, 260 és felette, pedig az elvárttól eltérően fog viselkedni, de érdekes eredményeket lehet vele elérni.

```
./imageproc -i input.ppm -o output.ppm --contrast 77
```



hue-shift

A színek HSL Hue értéket tolja el. Ajánlott értékei: -100 és +100 között.

```
./imageproc -i input.ppm -o output.ppm --hue-shift -11
```



invert

Invertálja a kép színeit.

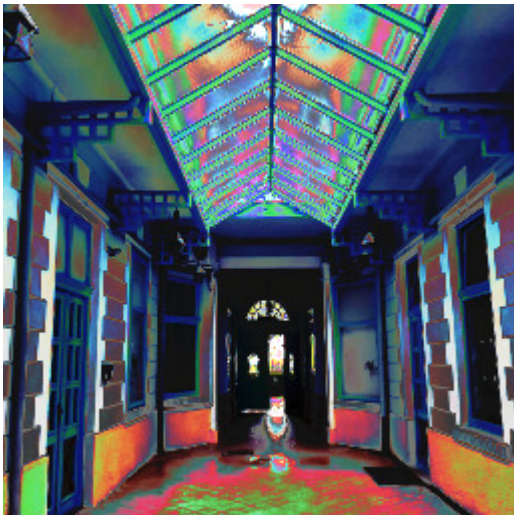
```
./imageproc -i input.ppm -o output.ppm --invert
```



sinecolor-shift

A kép színeinek értékét torzítja a szinusz függvény alapján. Ajánlott értékei: 0.004 és 0.05 között, vagy 10 és 50 között. Nagyon nagy érték zajos képet eredményez.

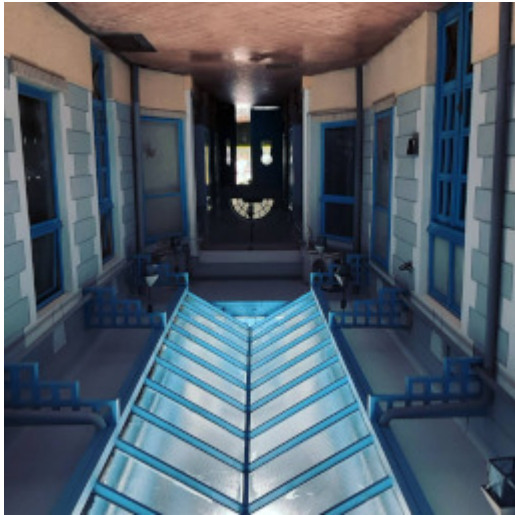
```
./imageproc -i input.ppm -o output.ppm --sinecolor-shift 0.013
```



mirror

Tükrözi a képet a megadott tengelyre. Lehetséges értékei: `vertical` , `horizontal` , `diagonal` , ahol sorrendben, a függőleges, vízszintes és bal felső-jobb alsó átlós tengelyre tükröz.

```
./imageproc -i input.ppm -o output.ppm --mirror diagonal
```

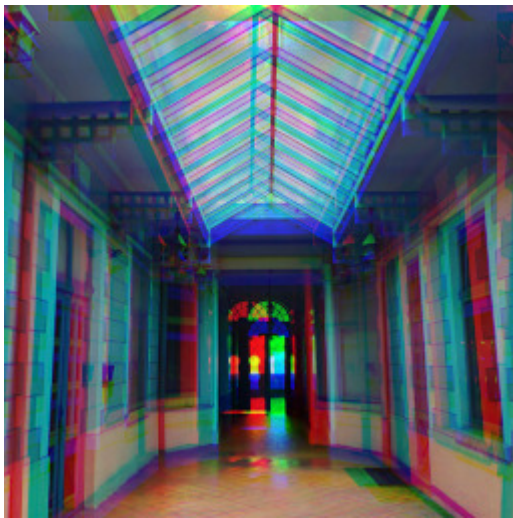


rgb-shift

A színcsatornákat csúsztatja el egyesével. Bemenet:

redx, redy, greenx, greeny, bluex, bluey , ahol sorrendben a vörös, zöld, kék csatornák elcsúsztatásának értékét kell megadni, először az x tengelyre, majd az y-ra. Ha nem akarunk egy csatornát csúsztatni, akkor is meg kell adni a 0 értéket. Ajánlott értékei maximum x irányban $\pm(\text{kép szélessége}/2)$, y irányban $\pm(\text{kép magassága}/2)$. Ennél nagyobb érték csak feleslegesen lassúvá teszi a programot.

```
./imageproc -i input.ppm -o output.ppm --rgb-shift -10,0,0,0,0,8 # a vörös  
csatornát vízszintesen eltolja 10 pixellel balra és a kékét 8 pixellel  
lefele
```

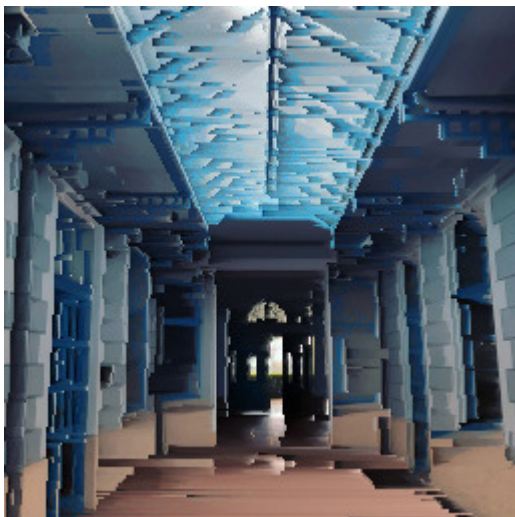


pixelsort

Rendezzi a pixeleket a megadott preset alapján. Lehetséges értékei:

- edges: megkeresi a kép objektumainak a szélét és ezek között rendez
- all-random: teljesen véletlenszerű beállítások
- landscape: tájképekhez és nagy tárgyakhoz
- macro: részletes képekhez használható
- fewcolors: kevés színt tartalmazó képekhez
- dark: sötét területek kiemelése

```
./imageproc -i input.ppm -o output.ppm --pixelsort edges
```



blur

Homályossá teszi a képet. Annyiszor fog végrehajtódni, amekkora megadott érték. Ajánlott értéke: maximum 100.

```
./imageproc -i input.ppm -o output.ppm --blur 1 # nagyon kicsi elmosás
```



sharpen

Élesebbé teszi a képet. Annyiszor fog végrehajtódni, amekkora megadott érték. Ajánlott érték: maximum 3. Ez az érték felett közelről szinte lehetetlen felismerni az eredeti képet. Csak akkor érdemes egynél nagyobb értéket használni, ha az eredeti kép homályos.

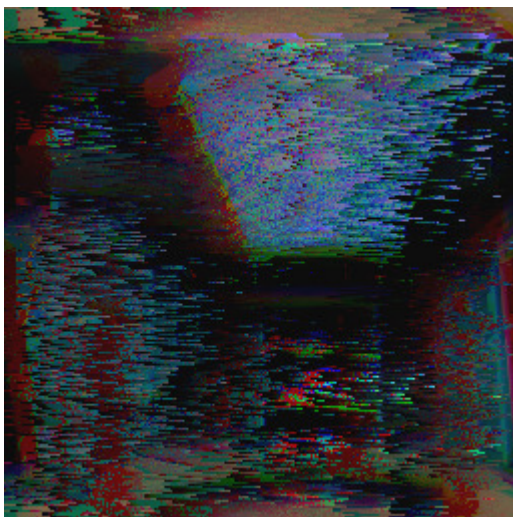
```
./imageproc -i input.ppm -o output.ppm --sharpen 1 # ennél nagyobb nem is  
értelmes használni
```



corrupt

Tönkreteszi a képet. Teljesen véletlenszerű, ezért mindig más eredményt ad. Néha teljesen felismerhetetlen a kép, máskor nem sok minden változik.

```
./imageproc -i input.ppm -o output.ppm --corrupt
```



grayscale

Fekete-fehérré alakítja a képet.

```
./imageproc -i input.ppm -o output.ppm --grayscale
```



3d

Vörös-cián 3D képet készít. A kép olyan hatású lesz, mintha be lenne süllyesztve.

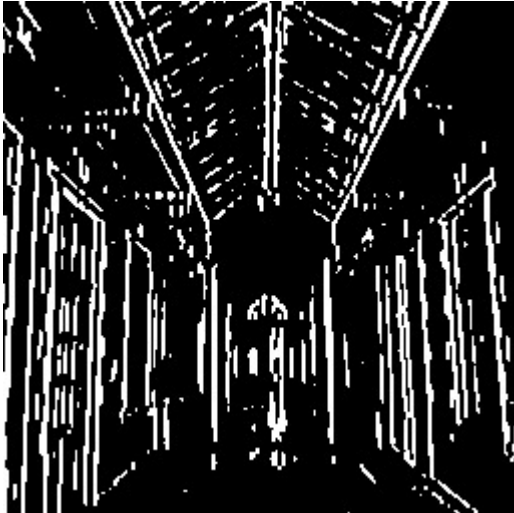
```
./imageproc -i input.ppm -o output.ppm --3d
```



edge-detect

Egy olyan fekete-fehér képet ad vissza, amin fehérrel látszanak a képen látható objektumok függőleges élei. Valódi haszna nincs, de segítségével ellenőrizhető, hogy a `--pixelsort edges` hol fog rendezni. Használatakor érdemes figyelembe venni, hogy a pixelsort a blur előtt van, tehát, amikor ezt használjuk, nem érdemes ennél kisebb prioritású műveletet elvégezni, mert a pixelsortot az nem érinti.

```
./imageproc -i input.ppm -o output.ppm --edge-detect
```



Összetett műveletek

A program természetesen több műveletet is képes egymás után végrehajtani, a felsorolt műveletek sorrendjében. A parancssori kapcsolók megadásának sorrendje ezt nem befolyásolja.

Példa 1. - Fekete-fehér 3D kép

```
./imageproc -i input.ppm -o output.ppm --grayscale --3d
```

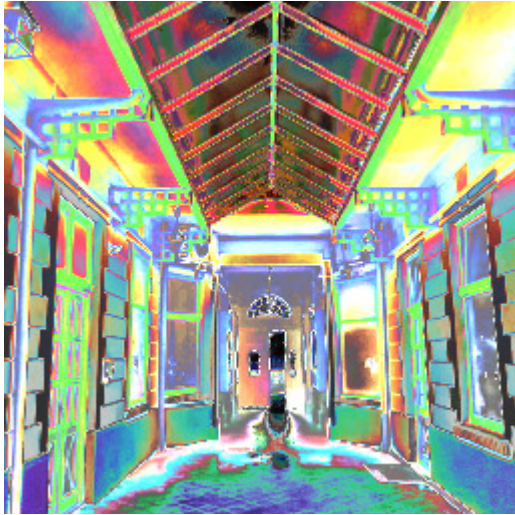
A képet először fekete-fehérré alakítja, majd utána 3D hatásúvá. Ez azért jó, mert vörös-cián 3D képnél nagyon zavaróak a színek, (főleg, ha van benne tiszta vörös, vagy kék), de a fekete-fehéren nem.



Példa 2. - Torzított színek

```
./imageproc -i input.ppm -o output.ppm --invert --sinecolor-shift 0.015
```

Invertáljuk a színeket, majd a sinecolor-shift segítségével ezeket a színeket torzítjuk.



Példa 3. - Fekete-fehér 3D kép módosított színekkel

```
./imageproc -i input.ppm -o output.ppm --invert --sinecolor-shift 0.015  
--grayscale --3d
```

Invertáljuk a színeket, majd a sinecolor-shift segítségével ezeket a színeket torzítjuk, végül ezt alakítjuk fekete-fehérré, majd majd 3D hatásúvá.



Példa 4. - Pixelsort dark presettel.

```
./imageproc -i input.ppm -o output.ppm --contrast 77 --pixelsort dark  
--blur 1
```

Előkészítjük a képet a pixelsortra, azzal, hogy megnöveljük a kontrasztot, így több sötét rész lesz a képen, majd végrehajtjuk a pixelsortot. A blur-re azért van szükség, mert a pixelsort miatt nagyon éles határok lesznek, a módosított és az eredeti pixelek között.



Példa 5. - Pixelsort edges presettel.

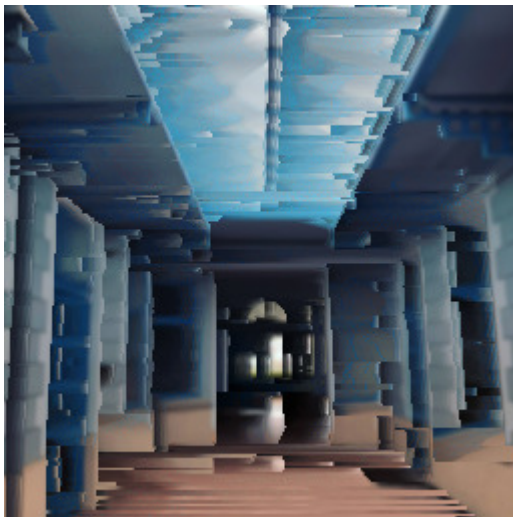
```
./imageproc -i input.ppm -o output.ppm --edge-detect
```

Megnézzük, hogy megfelelő számú él van a képen. Ha túl sok, akkor nagyon kis intervallumon lehet csak alkalmazni a pixelsortot. Ilyenkor lehet valamennyi blur-t használni, hogy kevesebb legyen.

```
./imageproc -i input.ppm -o input.ppm --blur 1
```

Ha így megfelelő a kép, akkor mehet a pixelsort.

```
./imageproc -i input.ppm -o output.ppm --pixelsort edges
```



Példa 6. - Komplex Glitch Art

```
./imageproc -i input.ppm -o output.ppm --sinecolor-shift 0.013 --pixelsort  
edges --blur 1 --sharpen 1 --3d --contrast -22
```

A sinecolor-shift torzítja a színeket. A torzított színek így más fajta edgeket eredményeznek, ezért a pixelsort kiszámíthatatlanul fog működni és egyedi eredményeket ad. A negatív kontraszt szaturáltabbá teszi a végeredményt. A blur-sharpen kombináció pixel-art típusú képet eredményez, mivel a blur esetén elvesznek az amúgy is nagyon különböző pixelértékek, amit a sharpen próbál visszanyerni, így egy zajosabb kép keletkezik. A 3d tovább módosítja a képet. Végül egy, az eredetitől teljesen különböző képet kapunk.

