

# Transformation de domaine

## 1 Interpolation bilinéaire

Implanter dans un module en C (`domain.c`) la fonction `unsigned short bilinear_interpolation(float x, float y, pnm ims, int c)` qui permet de faire une interpolation avec les quatres plus proches voisins.

## 2 `move.c`, `scroll.c` et `flip.c`

Écrire les programmes suivants :

1. `move <dx> <dy> <ims> <imd>` où `<dx>` et `<dy>` sont des entiers correspondant aux translations dans les directions  $x$  et  $y$ .
2. `scroll <dx> <dy> <ims> <imd>` où `<dx>` et `<dy>` sont des entiers correspondant aux translations dans les directions  $x$  et  $y$ .
3. `flip <dir> <ims> <imd>` où `<dir>`  $\in \{h, v, t\}$  est un caractère permettant de déterminer l'orientation de la transformation (i.e., `h` pour un flip horizontal, `v` pour un flip vertical et `t` pour une transposition). **Votre programme devra pouvoir accepter différentes combinaisons de paramètre, par exemple : `vht`, `hhh`, etc**

## 3 `shear.c` avec interpolation bilinéaire

Écrire le programme `shear <dir>\{h, v\} <angle> <ims> <imd>`, permettant de réaliser une transformation de type cisaillement dans le direction `<dir>` avec un angle `<angle>`. Votre programme devra utiliser l'interpolation bilinéaire.

## 4 `rotate.c` avec interpolation bilinéaire

Écrire le programme `rotate <x> <y> <angle> <ims> <imd>` qui utilise la méthode d'interpolation bilinéaire. Les paramètres `<x>` `<y>` correspondent à la position du centre de rotation et `<angle>` est un nombre flottant représentant l'angle de rotation exprimé en degrés.

La figure 2 montre également comment obtenir une rotation à partir de cisaillement. Tester que vos programmes peuvent également réaliser cette transformation.

## 5 Bonus : `sphere.c` avec interpolation bilinéaire et génération d'animation

Écrire un programme `sphere.c` qui permet de faire une projection d'une image sur une sphère. Si vous utilisez l'image de test (`earth.ppm`), la figure 1 montre un exemple de cette transformation géométrique.

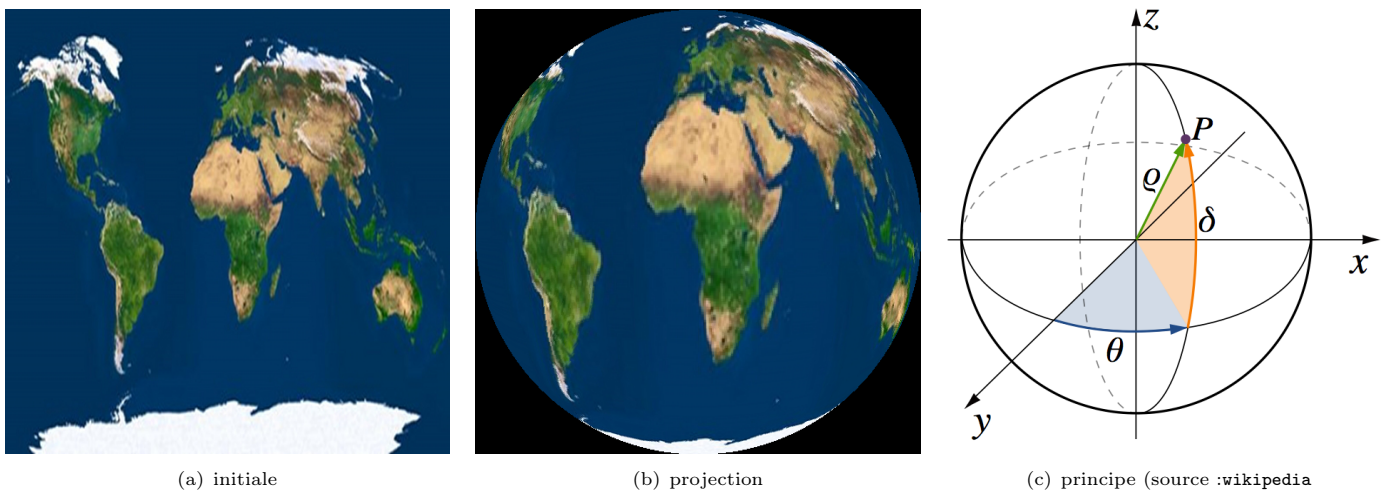


FIGURE 1 – Exemple de projection sur une sphère.

La figure 1 montre également le principe de cette transformation. L'idée générale est la suivante :

- se placer au centre de l'image
- pour tout les pixels vu comme un vecteur  $(y, x)$
- déterminer l'angle de projection

- déterminer la projection du vecteur au point  $P$
- calculer les coordonnées  $(\theta, \delta)$  dans la projection

Si vous générez une série d'images (par exemple à l'aide de `scroll`), une animation gif peut être créée en utilisant la commande `convert` avec les options `-delay` et `-loop`. Le fichier `globe.gif` est un exemple produit à partir de l'image de test.

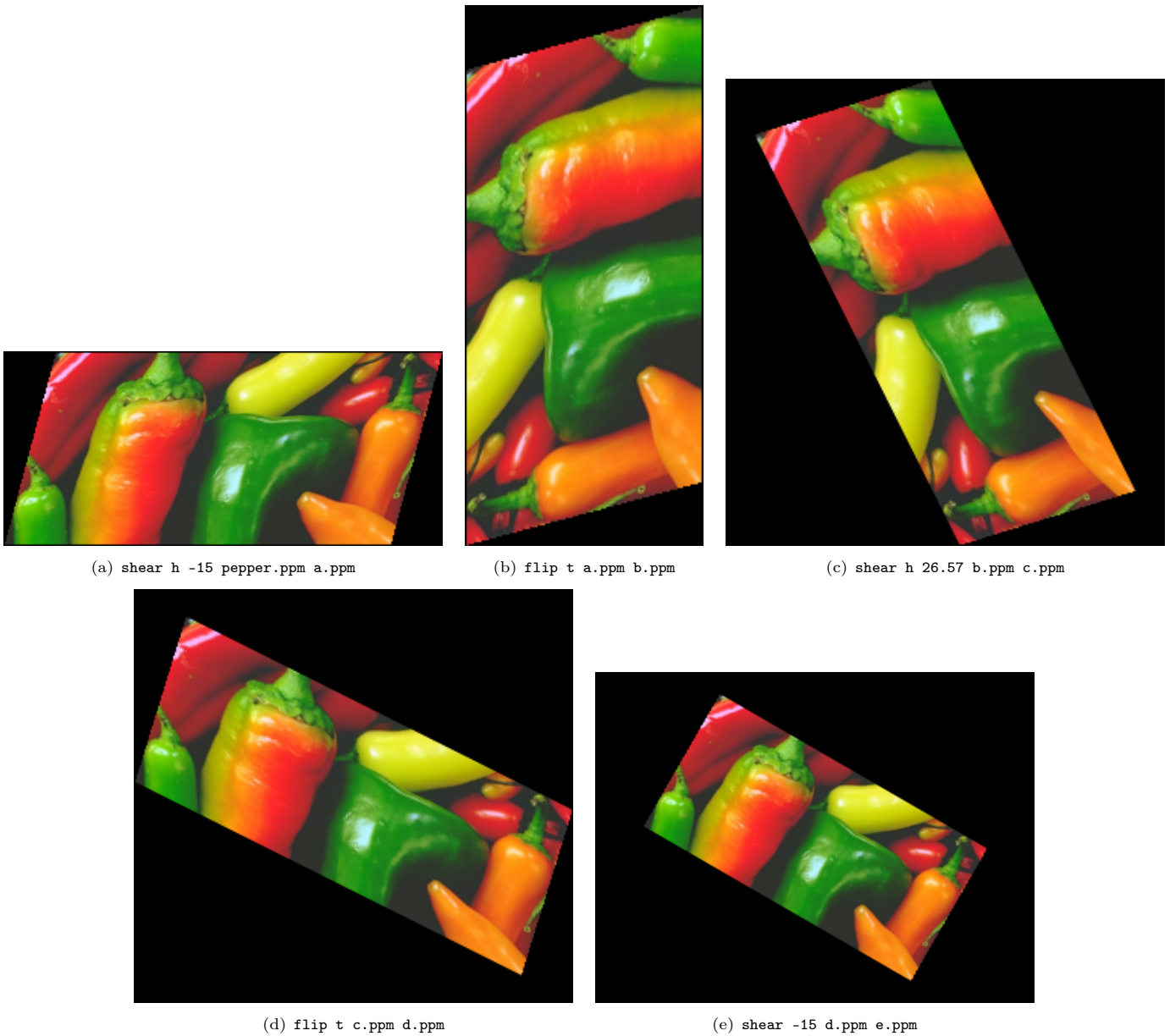


FIGURE 2 – Exemple de rotation d'angle de 30 degrés par succession de cisaillement et de transposition.

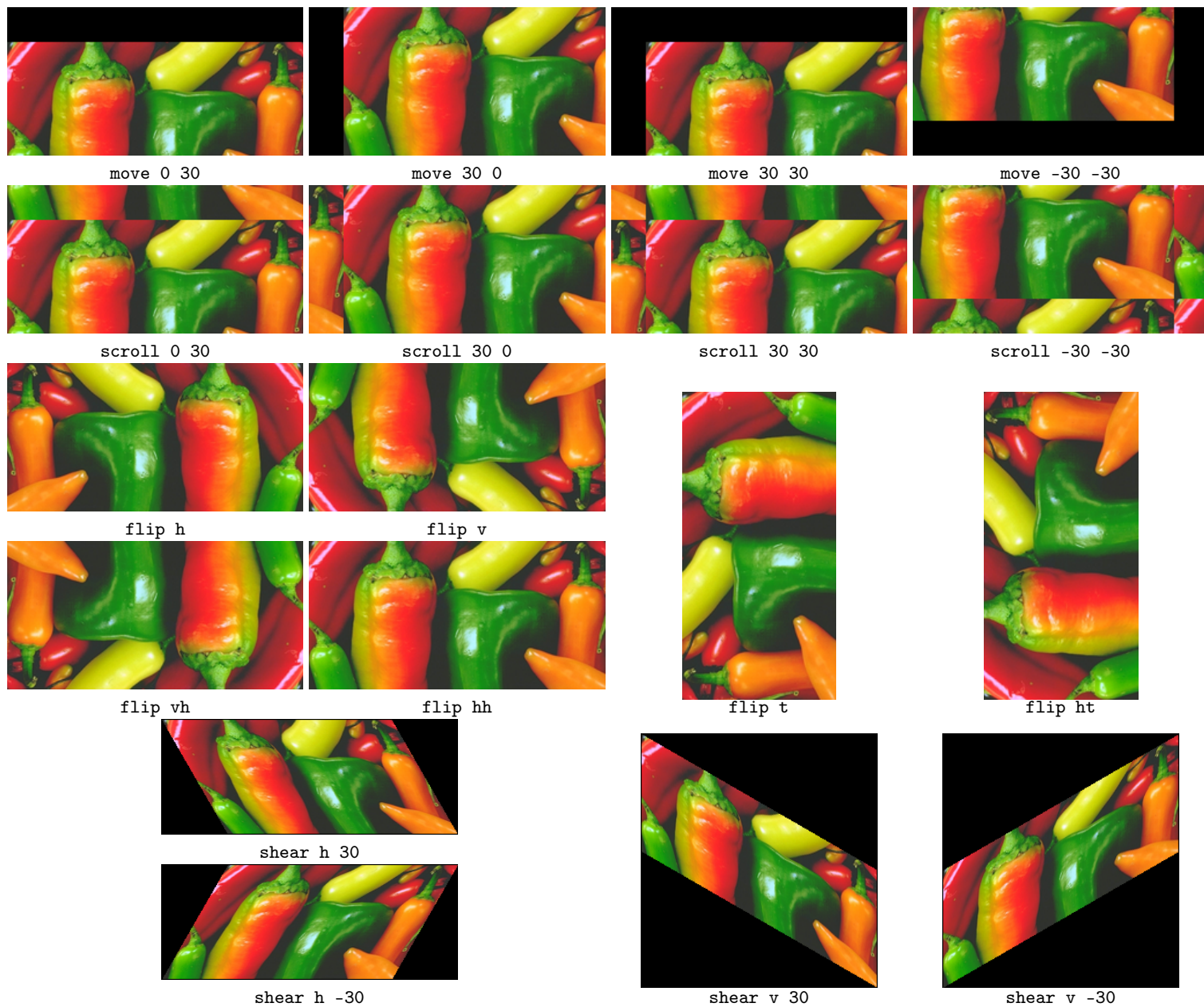


FIGURE 3 – Exemples d'exécution des programmes move, flip, scroll, rotate, et shear avec leurs paramètres d'exécution.