

# Proposition de TER - Création de mosaïque de photo «sous contrainte»

## Remarques préliminaires

*Les problèmes rencontrés et les oublis constatés dans ce support sont à me signaler à l'adresse [clot@univ-lyon1.fr](mailto:clot@univ-lyon1.fr).*

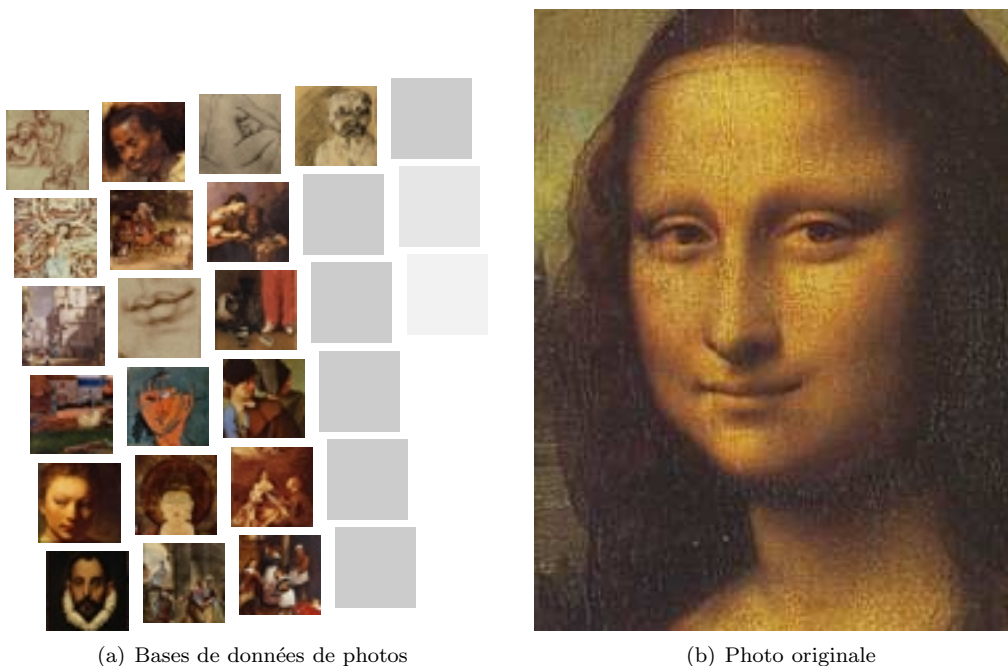
La proposition de TER détaillée ci-dessous constitue un prolongement de l'enseignement «Unix/Linux : interaction et programmation». Le but du TER est de favoriser un travail et une réflexion autour d'une problématique précise qui seront présentés sous la forme d'un rapport concis (les sources d'un travail de programmation devant également être transmises par un moyen permettant leur usage direct). La qualité de ce rapport tout comme celle du travail seront prises en compte pour l'évaluation.

Après un court scénario visant à présenter le problème pratique considéré, les caractéristiques de l'outil à construire seront présentées.

## Scénario

A partir d'une base de données d'images (cf Fig.1(a)) contenant plusieurs centaines de photos et d'une photo (ici, une extrait de la Mona Lisa de De Vinci, cf Fig.1(b)), nous cherchons à réaliser un assemblage régulier à partir des photos de la base de données comme celui proposé par la figure 3.

Pour cela, un découpage de la «cible» est réalisé (cf Fig.2), sous la forme d'une partition formée d'image de même tailles et de proportions identiques à celles des photos de la base (toutes les photos de la base sont supposées partager les mêmes proportions).



(a) Bases de données de photos

(b) Photo originale

FIG. 1 – Données initiales

Pour chaque image, il est possible d'obtenir les mesures suivantes :

- moyenne, variance, valeur minimum et maximum des pixels de l'image par rapport à la composante rouge
- idem pour la composante verte
- idem pour la composante bleue

A partir d'une distance prenant en compte ces informations, il est possible de déterminer pour chaque portion de la cible, la ou les images de la base de photos les plus proches au sens de la distance considérée. Ceci nous donne un moyen pour construire un assemblage.

Les outils permettant d'obtenir les mesures citées et de réaliser un découpage/montage d'image sont fournis par **ImageMagick**. Les commandes telles que **identify**, **mogrify** et **montage** devraient fournir la base relative aux manipulations d'images.

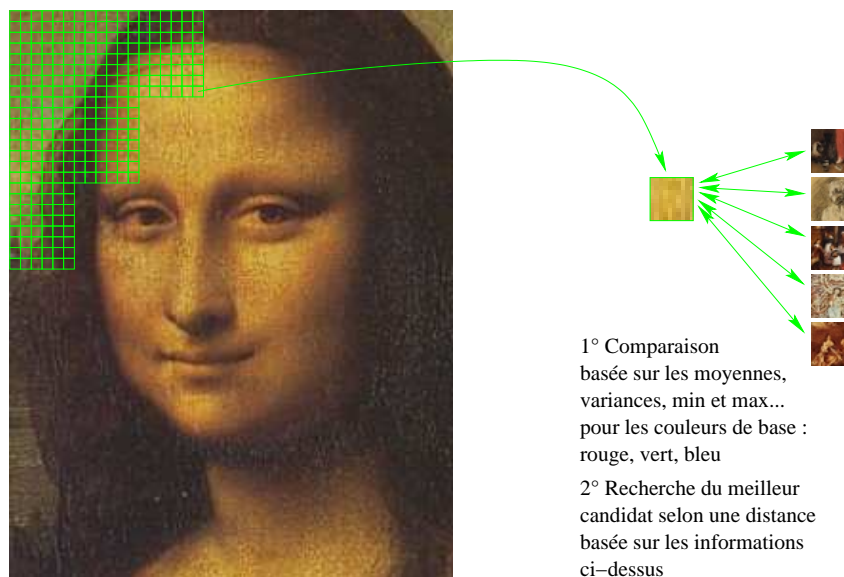


FIG. 2 – Decoupage et comparaison avec les photos de la base

Cette présentation un peu idéalisée a laissé de côté diverses considérations pratiques :

- Comment la taille de l'élément de la partition est-elle définie ?
- Comment gérer les cas pour lesquels la taille de l'élément de base ne conduit pas à un recouvrement parfait ?
- Est-il possible de rechercher de bons ajustement pour cette taille ?
- Comment tester la richesse de la base de données ?
- Et la distance ?
- ...

De cette présentation, nous pouvons dégager les principales fonctionnalités de l'outil de création de mosaïque.

## Fonctionnalités

L'outil de création de mosaïque sera un simple script reposant sur l'usage de commandes telles celles-mentionnées plus haut, cimentées par l'usage d'un shell choisi parmi ceux de la famille des bourne shells. Il n'est rien dit sur la base de données de photos. Il peut s'agir d'un simple répertoire contenant les photos, ou d'une autre structure. Ce détail est volontairement laissé libre. Rien n'est précisé sur la (ou les ?) distance, selon la même volonté.

Le programme construit disposera au moins des fonctionnalités suivantes :

- Affichage des informations relatives à la base de données (nombre de photos, information sur la qualité de l'ensemble...)
- Test des paramètres de découpage donnés à priori
- Construction d'une mosaïque, les paramètres de découpage étant donnés

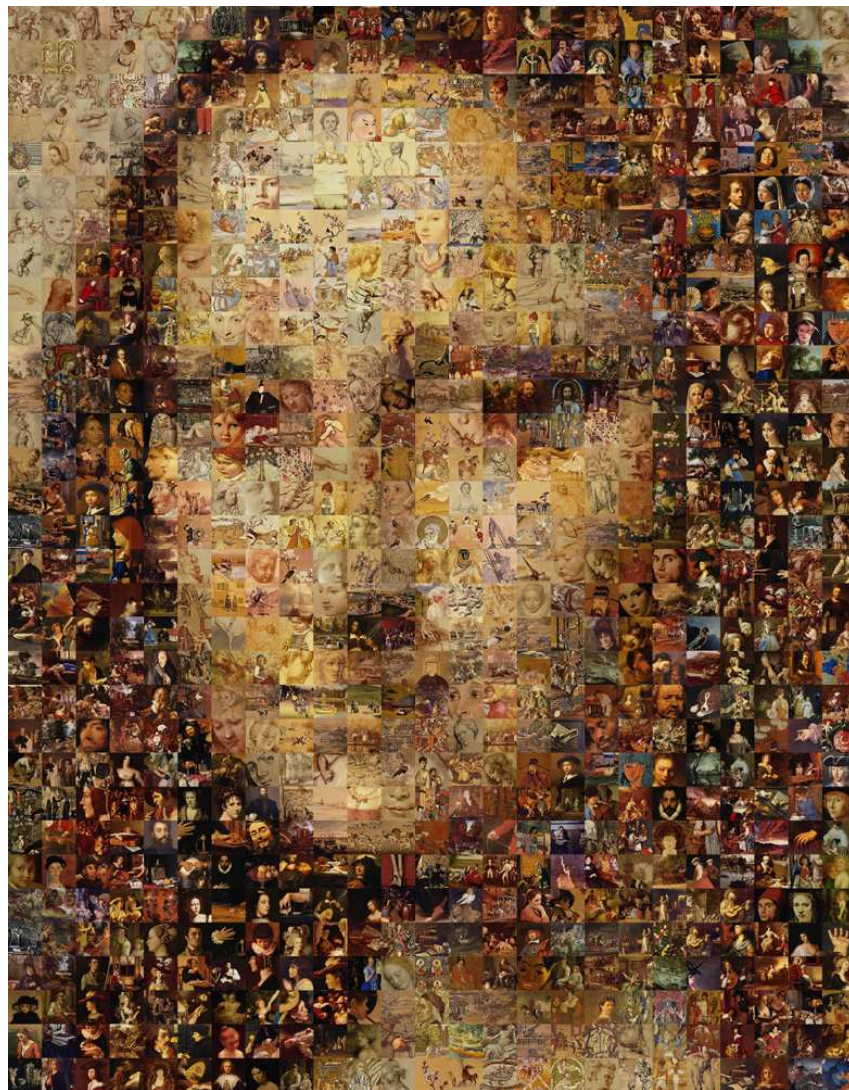


FIG. 3 – Exemple de mosaïque de photos

De nombreuses fonctionnalités s'ajouteront de façon naturelle en fonction des choix qui seront effectués.

Par ailleurs, les options classiques suivantes devront être implémentées :

- h permettra d'afficher une aide sur l'utilisation de la commande;
- o permettra de définir la sortie standard du programme;
- t permettra une exécution virtuelle, ie affichant le détail des opérations qui seraient exécutées en mode réel;
- v permettra d'invoquer un mode «verbeux».