

# Suivi de mouvements

## Application à la scénographie interactive et à l'installation interactive

Colin Bouvry - [colin.bouvry@laposte.net](mailto:colin.bouvry@laposte.net)

M1 ISIS – 2005/2006

UVHC – Département Audiovisuel et multimédia

Tuteur : Philippe Thomin

# Table des matières

1 RÉSUMÉ.....	3
2 INTRODUCTION.....	3
3 LES LOGICIELS D'INTERACTION TEMPS REEL.....	4
3.1 PURE DATA.....	4
3.2 PROCESSING.....	4
3.3 EYESWEB.....	4
4 CHOIX DE LA SOLUTION.....	5
4.1 SOLUTION 1.....	5
4.2 SOLUTION 2.....	5
4.3 SOLUTION 3.....	5
5 SOLUTION 1.....	6
5.1 Triangulation par relevé des distances.....	6
6 SOLUTION 2.....	6
6.1 Utilisation de processing .....	6
6.2 Utilisation d'eyesweb.....	7
6.3 Utilisation de pure-data.....	7
7 SOLUTION 3.....	8
7.1 PREMIER CAS.....	8
7.2 DEUXIEME CAS.....	9
7.2.1 Première solution.....	9
7.2.2 Deuxième solution.....	11
7.3 PROGRAMMATION .....	12
7.3.1 HYPOTHESE.....	12
7.3.2 CAPTATION.....	12
7.3.3 CALIBRATION.....	13
7.3.4 DETECTION.....	13
7.3.5 NUMEROTATION.....	15
7.3.6 SUIVI DE NUMEROS.....	16
7.3.7 COMMUNICATION.....	17
7.3.8 RECEPTION SUR PROCESSING.....	18
8 Ce qu'il reste à faire !.....	19
9 CONCLUSION.....	19
10 REFERENCES.....	20
11 ANNEXES.....	21
11.1 Utilisation de la bibliothèque OPENCV sous DevC++.....	21
11.2 Utilisation de la bibliothèque OPENCV sous VisualC++ 6.....	22
11.3 EXEMPLE DE CREATION INTERACTIVE MULTIMEDIA : Corps, graphisme et son.....	23
11.4 Chromakey.....	27

## **1 RÉSUMÉ**

Le but de ce projet est de donner à des artistes « audiovisuel » la possibilité de créer une composition sonore et/ou visuelle en temps réel à partir de la position, du déplacement de différentes personnes ou objets dans un espace de scène donné.

Ce système pourra être utilisé pour des installations interactives ou pour des spectacles vivants interactifs, voir en annexe « exemple de création interactive multimédia : Corps, graphisme et son ».

Le présent papier parle d'une étude des différents systèmes pouvant être utilisés et du développement d'une solution choisie.

## **2 INTRODUCTION**

Le projet proposé vise à la réalisation d'un système capable de reconnaître différentes positions, de suivre ces positions lors de déplacements de personnes dans un espace de scène. Ce système n'est pas censé différencier les différents acteurs selon leurs caractéristiques mais selon leurs positions initiales.

Le choix s'est porté sur un système de vision par caméra. Le système de vision utilisé sera réalisé par informatique et sera choisi dans une optique de compatibilité avec différentes plateformes et de liberté dans l'utilisation de bibliothèques ou de logiciels. Si la détection des acteurs s'avère difficile, les personnes en action pourront porter un élément sur eux permettant de mieux les détecter.

Le projet entend m'apprendre à choisir la bonne solution et à réaliser des tâches primaires comme le design et l'implantation d'un système simple de vision ayant des blocs d'acquisition, de traitement, et de production de résultats. Je vais essayer également d'analyser qualitativement et quantitativement le système de vision.

Je vais essayer durant tout le projet de respecter au mieux un compromis qualité du résultat par rapport à la vitesse de traitement.

Pour choisir le bon système, il est nécessaire de connaître les outils des artistes, ce qu'ils sont capables de faire et leurs performances. Ces outils sont nommés logiciels d'interaction temps réel ou logiciels de composition audiovisuelle.

### 3 LES LOGICIELS D'INTERACTION TEMPS REEL

Les logiciels et environnements de programmation en temps réel permettent de multiples applications artistiques, en concert mais aussi dans les représentations live.

Toutes les interactions peuvent être envisagées entre musiciens, danseurs, comédiens, et même spectateurs, pour imaginer des performances scéniques et des installations originales (sonores, multimédias, réalité virtuelle).

Il en existe plusieurs :

Propriétaires : MAX/MSP, ISADORA...

Libre : PURE-DATA, EYESWEB, PROCESSING...

Le choix s'est plus orienté vers les logiciels libres (licence publique).

#### 3.1 PURE DATA

Pure Data est un logiciel libre et ouvert (Windows-Linux-Mac OSX) développé par une grande communauté d'artistes et de programmeurs.

Il permet la manipulation ou la génération en temps réel de sons, images, vidéos, textes, flux de réseau, capteurs, actionneurs... tout médium numérique en entrée et en sortie.

Pure Data se dédie à tous les usages artistiques.

#### 3.2 PROCESSING

Processing est un langage de programmation et un environnement de développement créé par Benjamin Fry et Casey Reas, deux artistes américains. Processing est le prolongement "multimédia" de Design by numbers, l'environnement de programmation graphique développé par John Maeda au Media Lab du Massachusetts Institute of Technology.

Processing est tout particulièrement adapté à la création plastique et graphique interactive. Le logiciel fonctionne sur Macintosh, sous Windows et sous Linux, car il est basé sur la plateforme Java. Il permet d'ailleurs de programmer directement en langage Java.

Processing est distribué sous licence GNU.

#### 3.3 EYESWEB

Eyesweb est un environnement logiciel permettant de gérer des applications multimédias temps réel grâce à un langage de programmation graphique (patches). Il est très performant pour la détection de paramètres sur des danseurs.

## **4 CHOIX DE LA SOLUTION**

Différents systèmes peuvent être envisagés

### **4.1 SOLUTION 1**

Calcul de la position avec un badge émetteur électronique porté par chaque personne évoluant dans un espace de scène. Trois récepteurs transmettant la distance des émetteurs à un central. Un central transmettant en temps réel à une unité informatique la distance de chaque émetteur individuel.

**Inconvénient :**

Difficile à mettre en oeuvre.

Nécessité du port d'un badge électronique.

### **4.2 SOLUTION 2**

Choisir le logiciel de composition audiovisuelle et élaboration d'un patch pour calculer la position à partir d'une image vidéo. L'artiste composera sa création audiovisuelle sur ce logiciel ou sur un autre à partir de données recueillie par ce patch.

**Avantage :**

Facilement compilable sur différentes plates-formes.

Bénéficie de nombreux patch.

**Inconvénient :**

Limité par les fonctions du logiciel.

Programmation de nouvelles fonctions assez difficile.

### **4.3 SOLUTION 3**

**Analyse vidéo :**

Captation par caméra placée au dessus d'un sol différenciateur dans un espace de scène.

Elaboration d'un programme d'analyse de type c++ restituant les coordonnées de chaque personne. Communication en temps réel à un logiciel de composition sonore et visuel les positions.

**Avantage :**

Puissance du c++.

## **5 SOLUTION 1**

Il est possible de déterminer les positions de personnes avec un émetteur sur chaque personne, plusieurs récepteurs et un ordinateur qui calcule et communique les positions. Cette solution peut être utilisée avec des ultrasons ou des ondes électromagnétiques.

### **5.1 Triangulation par relevé des distances**

On peut relever une position en estimant la distance par rapport à trois points. Si l'on prend deux points de référence, on a deux sommets et les longueurs des côtés ne joignant pas ces sommets, ce qui définit deux triangles ; le point du relevé se trouve au troisième sommet de l'un de ces triangles. L'adjonction d'un troisième point de référence permet de déterminer lequel des deux sommets est le bon.

#### **Utilisation de l'intensité d'un signal**

Avec une onde électromagnétique, on peut utiliser l'intensité du signal collectée par une antenne non directionnelle. Si le milieu de propagation est homogène et isotrope, l'intensité diminue selon le carré de la distance (l'énergie se répartit sur une sphère grandissante). L'intensité permet donc d'estimer la distance, et donc de situer l'émetteur sur un cercle centré sur le récepteur. Un deuxième récepteur permet de tracer un second cercle, l'émetteur se trouve donc à l'intersection des deux cercles ; un troisième récepteur permet de déterminer lequel des deux points d'intersection est le bon (ou bien la logique, un navire ne peut pas se trouver sur les terres).

#### **Utilisation de la vitesse de propagation d'un signal**

Si l'on a des événements émettant un signal, alors en constatant le décalage dans l'arrivée des signaux, on peut déterminer la différence de distance entre les événements et les récepteurs, à condition de connaître la vitesse de propagation du signal.

Si le récepteur est lui-même synchronisé avec les émetteurs, on peut alors déterminer directement le temps de trajet et donc la distance entre émetteurs et récepteur.

#### **Remarque :**

Il faut savoir identifier les différents utilisateurs. Il est donc nécessaire de coder chaque émetteur pour pouvoir les différencier.

#### **Inconvénient :**

Équipement assez lourd et coûteux.

## **6 SOLUTION 2**

### **6.1 Utilisation de processing**

Pour que processing puisse utiliser de la vidéo d'une webcam, il faut d'abord installer la bibliothèque JMYRON. La captation de la vidéo nécessite quicktime avec les plugins suivants « quicktime for java » et « quicktime capture », et le logiciel VDIG qui permet de capter la vidéo d'une caméra, de faire des réglages de la captation et de communiquer les informations au format quicktime.

## Problèmes rencontrés :

La captation est trop longue, le taux de rafraîchissement de la caméra est insuffisante. (testé sur windows). Peu de caméras sont compatible avec le logiciel VDIG. Un autre logiciel « abstract plane VDIG » existe présentant une meilleure compatibilité avec les caméras mais payant.

Cette solution est abandonnée.

## 6.2 Utilisation d'eyesweb

De nombreuses bibliothèques de fonctions sont en utilisation libre comme EyesWeb Motion Analysis Library mais certaines sont payantes comme EyesWeb Trajectory Analysis Library ou EyesWeb Space Analysis Library.

La bibliothèque EyesWeb Motion Analysis Library est limitée par les fonctions proposées pour répondre à l'objectif.

Eyesweb permet de développer rapidement sous forme de patch par rapport au c++.

Le développement d'une fonction sur eyesweb est complexe.

## 6.3 Utilisation de pure-data

L'illustration ci-dessous est un exemple pour suivre des mouvements sur pure-data avec la bibliothèque graphique GEM. Source venant de « **Pure Data** : network and Applications » de SM 1205 Interactivity.

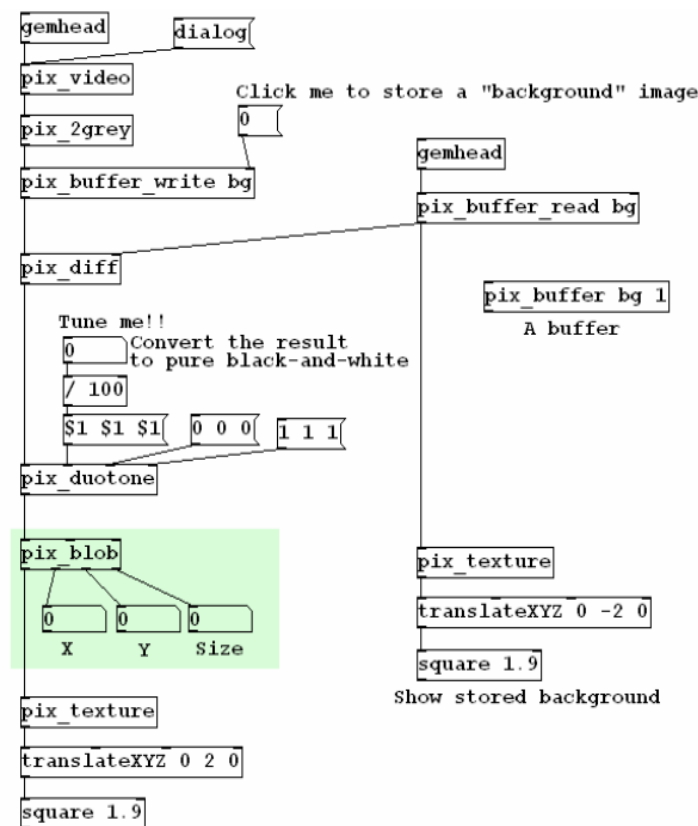


Illustration 1: Suivi de mouvements avec GEM

**pix\_2grey** : converti RGB pixels en niveaux de gris  
**pix\_diff** : calcul la valeur absolu de la différence entre deux pixels  
**pix\_buffer\_write** : met les pixels dans pix\_buffer  
**pix\_buffer\_read** : charge les pixels deus pix\_buffer  
**pix\_duotone** : réduit le nombre de couleur de la binarisation  
**pix\_blob** : calcul le centre de gravité des blobs

### **Problèmes rencontrés :**

Difficulté d'installer les bibliothèques nécessaires et de compiler sur windows.

## **7 SOLUTION 3**

Tout d'abord, il est possible que l'acteur sur la scène porte quelque chose sur lui permettant de le distinguer plus facilement.

### **Plusieurs approches sont possibles :**

#### **Premier cas :**

L'utilisateur ne porte rien de particulier sur lui qui permet de l'identifier.

#### **Deuxième cas :**

L'utilisateur a quelque chose de particulier sur lui qui permet de l'identifier.

Par la suite, un programme écrit en c++ tente d'identifier les positions selon les deux cas possibles.

### **7.1 PREMIER CAS**

L'acteur ne porte pas d'identifiant sur lui. Cela pose de nombreux problèmes de détections des danseurs. En effet, les acteurs peuvent avoir des couleurs différentes, multipliant ainsi le nombre de cas possibles. L' image de gauche ci-dessous montre une personne avec des vêtements sombres et des chaussettes blanches. L'image de droite montre deux personnes portant des couleurs vives qui se chevauchent. Ces situations sont difficiles à traiter.



*Illustration 2: Exemples d'acteurs en mouvement*



## 7.2 DEUXIEME CAS

Plusieurs identifiants sont possibles. L'étude se porte sur deux types d'identifiants. La première solution se repose sur un programme pouvant répondre à l'objectif qui utilise un certain identifiant. La deuxième solution utilise un identifiant lumineux ou réfléchissant.

### 7.2.1 Première solution

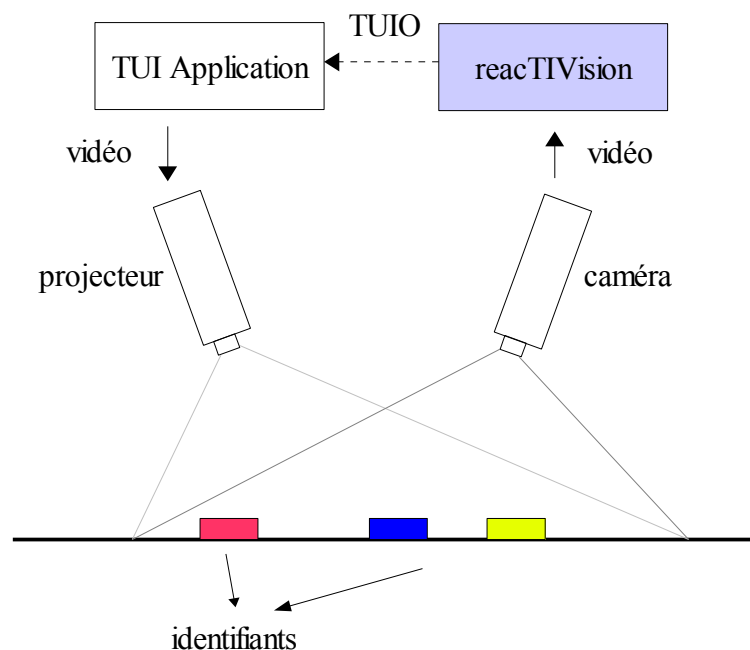
Un identifiant est placé sur la tête d'une personne. Cet identifiant peut être placé sur un chapeau. L'identifiant permet de suivre une personne en mouvement tant qu'il est bien visible. L'identifiant est dessiné en noir et blanc de telle sorte que le contraste soit au maximum. La combinaison des dessins est réalisée de telle sorte qu'elle soit très peu probable de la retrouver dans une scène quelconque.

La reconnaissance de l'identifiant est invariante :

- aux différences de couleurs et de luminosité de l'éclairage de la scène
- au grossissement de l'identifiant
- à la rotation de l'identifiant
- à l'inclinaison de l'identifiant jusqu'à un certain seuil de tolérance

Le logiciel en question qui est capable de faire cela est reacTIVision.

ReacTIVision est un système « open source » pour suivre la position et l'orientation de marqueurs sur de la vidéo en temps réel. Ce logiciel a été créé par Costanza et Robinson's. Ce logiciel est réalisé en c++ et compilé sur plusieurs plates-formes. Il communique les données calculées en protocole OSC. De nombreux exemples avec les logiciels visuels et sonores sont disponibles pour exploiter les données.



*Illustration 3: principe de reacTIVision*

### Exemple d'application :

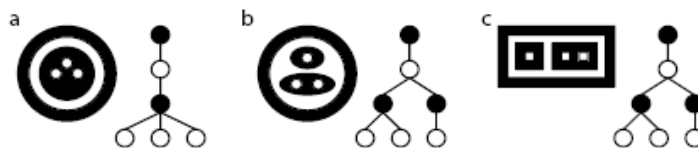


*Illustration 4: reactIVision en action*

### Explication des identifiants :

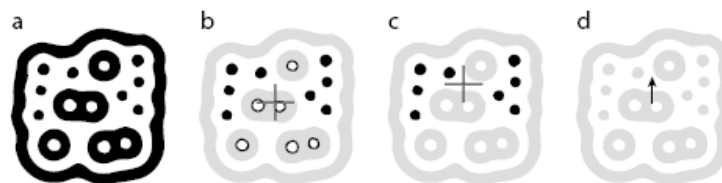
Il existe une multitude d'identifiants tous différents mais construits d'une manière à pouvoir les reconnaître, les différencier, leur calculer la position et l'angle de rotation.

Les figures suivantes présentent différentes topologies simples avec leurs graphes de régions correspondantes. La source vient de « The Design and Evolution of Fiducials for the reactIVision System ».



*Illustration 5: Topologies et leurs graphes de régions*

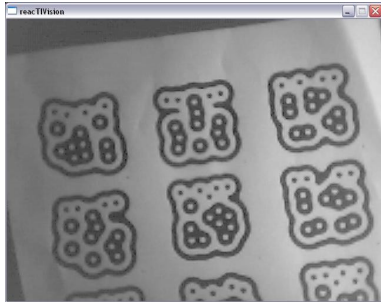
La figure (a) est un identifiant de reactIVision. La figure (b) représente le centre de gravité calculé avec la moyenne des centres de gravité des points noir et blanc. La figure (c) représente l'ensemble des points noirs permettant de calculer un vecteur (d) entre les centres de gravité de ces points et de l'identifiant.



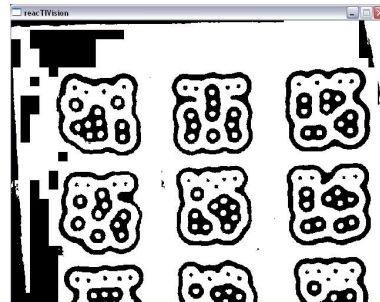
*Illustration 6: Principe du vecteur orientation*

### Test :

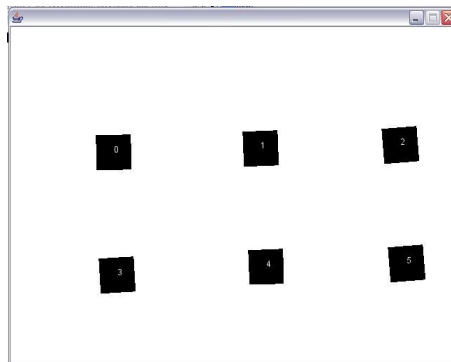
Le test a été réalisé avec une webcam quicktime messenger en basse résolution. La réception de la position des symboles a été testée en programmation java sur l'environnement processing.



*Illustration 7: Affichage simple*



*Illustration 8: Après binarisation*



*Illustration 9: Affichage de carrés en fonction des paramètres trouvés*

### Avantage :

Calcule la position et l'angle.  
L'identifiant de la personne est bien connu.  
Fonctionne avec différents éclairages.

### Inconvénient :

Nécessite une caméra avec une bonne résolution.

### 7.2.2 Deuxième solution

Il est possible d'utiliser un objet très lumineux de couleur, positionné sur une personne dans un environnement peu éclairé. La détection peut être facilitée avec un algorithme de type « chromakey ».



*Illustration 10: Objet lumineux*

Il est aussi possible d'utiliser un objet réfléchissant dans les infrarouges avec une caméra sensible aux infrarouges et un projecteur infrarouge.

### **7.3 PROGRAMMATION**

Captation par caméra placée au dessus d'un sol différenciateur dans un espace de scène. Le programme est écrit en c++. Le traitement d'image est réalisé avec la bibliothèque « opencv » en licence GPL et multiplateforme.

#### **7.3.1 HYPOTHESE**

- Pour le premier cas, les images sont captées dans un certain contexte pour simplifier la détection. Le test est réalisé sur une source vidéo au format « avi » de boules de billard. Les objets en mouvement sont de couleurs uniforme.
- La programmation réalisée est valable pour les deux cas : avec ajout d'identifiants et sans identifiants

#### **7.3.2 CAPTATION**

Pour facilité le développement, il est utile de pouvoir lire de la vidéo. Opencv permet de lire de la vidéo au format avi et de lire de la vidéo en temps réel venant d'une caméra. L'exemple ci-dessous illustre la lecture et l'affichage de la vidéo avec opencv.

##### **Captation à partir d'une webcam :**

```
CvCapture* capture=cvCaptureFromCAM(0);
if(capture)
{
    IplImage* frame=cvQueryFrame(capture);
}
```

##### **Captation à partir d'une source vidéo avi :**

```
CvCapture * capture = cvCaptureFromFile(filename);
if(capture)
{
    IplImage* frame=cvQueryFrame(capture);
}
```

## Affichage d'une image :

```
cvNamedWindow("Affichage image", 0);  
cvShowImage("Affichage image", frame);
```

### 7.3.3 CALIBRATION

La calibration permet de régler un appareil par rapport à des données de références afin d'obtenir un comportement fidèle lors de la reproduction.

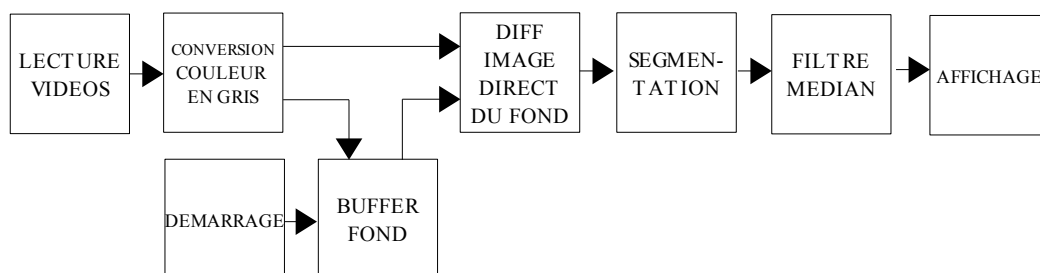
La qualité des résultats dépend directement de la position et des caractéristiques de la caméra.

Il est possible d'estimer les caractéristiques d'une caméra à partir de ses paramètres extrinsèques et intrinsèques. Si la caméra est bien positionnée, il n'est pas forcément nécessaire de calibrer. Pour gagner du temps, l'étape de calibration peut se faire sur les coordonnées des positions des personnes et pas sur l'image.

### 7.3.4 DETECTION

On peut éventuellement fixer la région que l'on veut traiter à l'aide de la fonction d'OpenCV « `setImageROI()` ».

Qu'est ce qu'une soustraction de l'image de fond ?



*Illustration 11: Soustraction de l'image de fond*

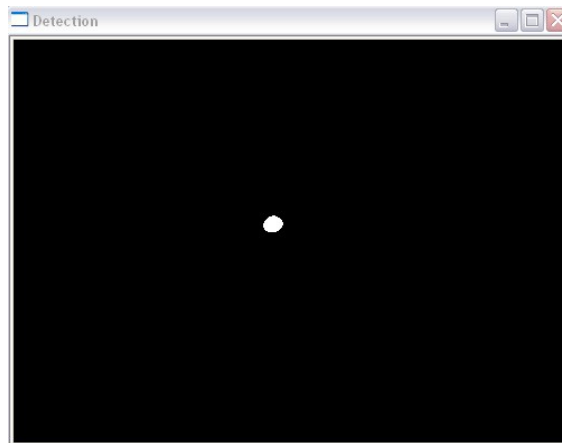
C'est une simple technique de détection de mouvement dans une scène statique.

Le principe est simple :

$$\text{mouvement} = \text{abs}(\text{image}[I] - \text{referenceImage}) > \text{seuil}$$

L'image de référence est enregistrée au démarrage de l'application dans un buffer.

La différence entre l'image courante et l'image de fond donne une information de mouvement. Cette différence est ensuite seuillée pour avoir une image binaire afin de bien identifier les parties en mouvement. Le filtre médian permet de supprimer des faux pixels. L'illustration ci-dessous permet de visualiser une détection de mouvement après le filtre médian.



*Illustration 12: Détection de mouvement*

### **Programme Opencv :**

```
cvAbsDiff(gray,fond,diff) ou cvSub(gray,fond,diff);
cvThreshold(diff, bin , 50, 255, CV_THRESH_BINARY);
cvSmooth(bin,median,CV_MEDIAN,5,5);
```

### **Explication :**

Cependant, cette méthode fonctionne bien quand le contraste entre le premier plan et le fond est très haut (noir contre le blanc, par exemple). Il est également important que les conditions de lumières ne changent pas trop et ne soient pas trop rapides. Une autre chose importante est d'éviter la présence des ombres qui peuvent être confondues avec des parties du corps du danseur.

Ce simple algorithme n'est pas très efficace car il utilise un seuil fixe sur toutes les parties de l'image alors qu'il vaudrait mieux changer ce seuil dans les zones qui sont plus foncées ou plus clair.

Il est possible de résoudre le problème avec un seuil par bloc d'image, afin d'obtenir une « silhouette » de la personne mobile. La fonction « cvAdaptiveThreshold » permet de s'adapter à la luminance selon un calcul de type gaussien ou moyenne.

### **Problèmes rencontrés :**

Si l'objet en mouvement a la même luminosité que le fond mais pas la même couleur, le processus ci-dessus ne fonctionne pas. Il est peut être possible de résoudre ce problème avec le principe de « chromakey » sur le fond ou sur l'objet en mouvement (Voir annexe « chromakey »). Le résultat d'extraction de couleur n'est pas satisfaisant. Une autre solution est peut être possible avec un algorithme d'extraction de contours.

### 7.3.5 NUMEROTATION

Il est ensuite utile d'attribuer pour chaque région (blob) trouvée un numéro.

Pour avoir une cohérence dans la numérotation des objets ou personnes en mouvements sur plusieurs images, il est nécessaire que les mêmes objets ou personnes aient le même numéro du début jusqu'à la sortie du champ de vision de la caméra. Sauf cas particulier où plusieurs blobs se croisent et en forme plus qu'un, dans ce cas lors de leurs séparations une réattribution aléatoire sera effectuée entre leurs numéros d'identifiant précédents.

La fonction « cvFloodFill » permet d'identifier une région connexe par sa position, sa taille et sa surface. Il est ensuite possible d'attribuer un numéro d'identifiant sur chaque région. Cette fonction sert en principe à colorier une région connexe.

Pour aller un peu plus loin dans l'étude des différentes régions, il est nécessaire d'implémenter un autre algorithme.

Il existe une multitude d'algorithmes d'étiquetage, celui retenu est l'étiquetage séquentiel avec correspondance entre points pour sa robustesse et sa rapidité.

#### Etiquetage séquentiel avec correspondance entre points

D'après le livre : « Analyse d'images : filtrage et segmentation » de [Cocquerez](#)

L'idée de l'algorithme est d'exploiter l'étiquetage effectué lors du premier parcours pour une affectation finale des étiquettes ne demandant ainsi qu'un seul parcours supplémentaire. Une table de correspondance  $T$  est créée et initialisée par  $T(i) = i$ . Le premier balayage séquentiel de l'image est défini par le traitement suivant appliqué en tout point  $P$

```
Si Tous les prédécesseurs de  $P$  appartiennent au fond
  Alors Affecter une nouvelle étiquette à  $P$ 
Sinon
  Si Tous les prédécesseurs de  $P$  qui sont objet ont une même étiquette
    Alors Attribuer cette étiquette à  $P$ 
  Sinon
    Rechercher la plus petite étiquette  $e \neq 0$  de ces prédécesseurs
    Affecter  $T(e)$  à  $P$ 
    Mettre à jour la table  $T$  par
    Pour chaque prédécesseur d'étiquette  $a$  telle que  $T(e) \neq T(a)$ 
      Faire
        Tant que  $T(a) \neq a$ 
          Faire
             $k = T(a)$ 
             $T(a) = T(e)$ 
             $a = k$ 
          Fin Faire
        Fin Pour
      Fin Si
    Fin Si
```

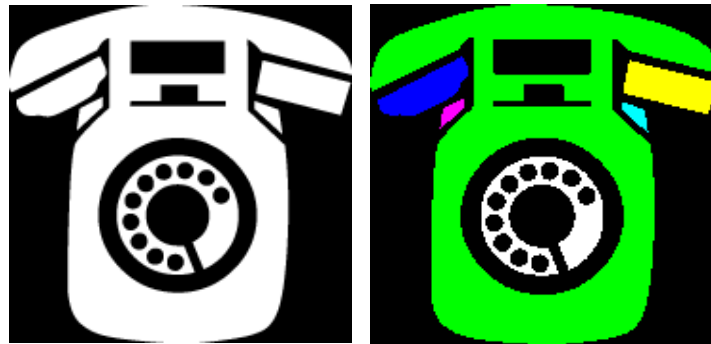
A la fin de ce balayage de l'image, on actualise la table  $T$  de manière à ce qu'à tout indice corresponde l'étiquette définitive de l'objet.

```

Pour i = 1 à nombre d'étiquettes utilisées
Faire
    j = i
    Tant que T(j) != j Faire j = T(j)
    T(i) = j
Fin pour

```

Au terme de ce traitement, les points d'un même objet peuvent avoir des valeurs différentes mais la table  $T$  permet de toutes les faire correspondre à une même étiquette. Le second balayage sert à l'attribution définitive sur l'image d'une même étiquette pour tous les points d'un même objet. Cet algorithme nécessite donc deux balayages et l'emploi d'une table de correspondance. Il induit également la génération de nombreuses étiquettes temporaires dans le cas d'un objet de forme complexe, ce qui implique la définition d'une table  $T$  de grande taille. Ceci est dû à la vue uniquement locale de l'algorithme lors de l'attribution des étiquettes.



*Illustration 13: Recherche des composantes connexes*

La figure ci-dessus illustre le résultat de cette recherche. Les pixels blancs (255=forme) de l'image de gauche sont rassemblés par "paquets" (composantes) connexes et à chaque composante connexe correspond une couleur (non noire) dans l'image de droite.

### 7.3.6 SUIVI DE NUMEROS

Des régions parasites peuvent s'introduire dans l'image. Il est possible d'enlever ces régions en calculant leurs caractéristiques. Différentes caractéristiques peuvent être calculées comme la surface, les couleurs moyennes RVB, l'écart type RVB, la luminance moyenne...

Une fois les régions filtrées, il faut que ces mêmes régions aient le même identifiant au cours du temps. L'étiquetage se faisant sur une image à la fois. Un label servant d'identifiant sera donc attribué sur chaque numéro de blob. Les paramètres permettant de définir qu'une région est la même sur deux images consécutives sont la distance inter-régions par rapport au centre de gravité et le nombre de pixel commun avec les autres régions.



### **Plusieurs cas particuliers peuvent se présenter :**

Tout d'abord la vitesse est trop grande par rapport à la fréquence de rafraichissement de l'image. Dans ce cas, une région sur deux images consécutives n'a plus de pixel commun. La décision devient alors difficile car la confusion peut se faire avec une autre région.

Ensuite, admettons que la vitesse par rapport à la fréquence de rafraichissement est bonne. Si plusieurs régions se rencontrent, les régions sont fusionnées pour en former plus qu'une. Le label le plus faible est mémorisé par défaut, les autres labels sont mémorisés et seront réattribués lors de leurs séparations. Les labels mémorisés ne pourront pas être attribués si une nouvelle région apparaît dans l'image. Une séparation est détectée quand la nouvelle région sort de la région de l'image précédente s'est à dire quand la nouvelle région a des pixels en commun avec la région de l'image précédente. Dans ce cas, le label est attribué aléatoirement entre les régions mémorisées.

Si une région disparaît, le label de cette région est libéré et une nouvelle région peut prendre ce numéro.

### **Problème rencontré :**

Difficulté à traiter tous les cas possibles.

## **7.3.7 COMMUNICATION**

Le but est de communiquer en temps réel à un logiciel de composition sonore et visuel les positions des objets ou personnes en scène.

Cette communication permet de séparer la partie détection de mouvement de la partie artistique. Cette séparation permettra la compatibilité avec n'importe quel logiciel de composition et permettra aussi de dédier un ordinateur pour chaque logiciel afin d'augmenter leurs rapidités d'exécution.

Il existe plusieurs protocoles de communication par réseaux, compatibles avec la plupart des logiciels d'interaction temps réel, comme l'OSC, le RTP MIDI et l'Ethernet.

Le protocole OSC (Open Sound Control) permet de communiquer facilement, gratuitement, et efficacement avec n'importe quels logiciels.

Le choix s'est porté sur la bibliothèque OSCpack.

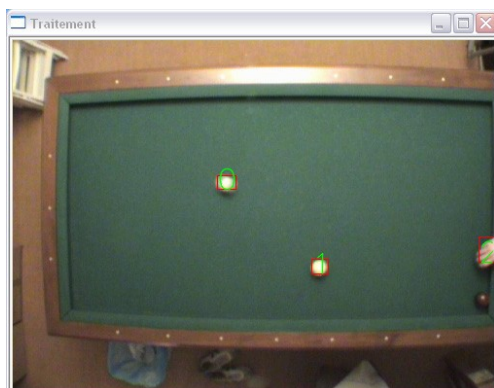
### **Description de OSCpack :**

Oscpack est un ensemble de classes C++ de paquets OSC. Oscpack inclut également une petite bibliothèque de classes de gestion de réseau d'UDP pour Windows et POSIX (Linux, OSX) qui sont suffisantes pour écrire beaucoup d'applications et serveurs OSC.

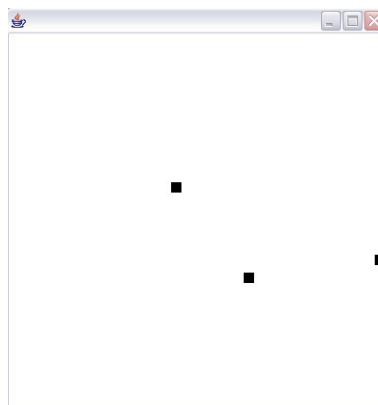
Une adresse IP, le numéro de port et l'adresse OSC sont configurés avant d'envoyer les informations. Après avoir détecté tous les blobs voulus dans une image, le centre de gravité du blob, normalisé par rapport à la dimension de l'image traitée, est envoyé sous l'adresse « /blob ». Une fois les coordonnées des blobs de l'image envoyés, un état booléen est envoyé sous l'adresse OSC « /blobSynch » pour dire que l'envoi des blobs d'une image est terminé.

### 7.3.8 RECEPTION SUR PROCESSING

oscP5 est une bibliothèque OSC pour la programmation sur l'environnement processing. L'illustration 15 présente un exemple d'affichage de la partie détection et transmission écrit en C++. L'illustration 14 présente l'affichage de carrés qui se déplacent suivant les coordonnées reçues par le protocole OSC.



*Illustration 15: Détection et transmission (OSC) des coordonnées*



*Illustration 14: Réception (OSC) des coordonnées*

#### Utilisation des positions :

Sur l'environnement processing, il est possible de calculer la vitesse, l'accélération et la trajectoire des boules de billard afin d'être utilisé pour déclencher un événement en temps réel. Par exemple, la position en abscisse d'une boule peut servir à varier la fréquence (pitch) d'un son et la position en ordonnée peut servir à varier le volume sonore d'un son.

## **8 Ce qu'il reste à faire !**

Il reste à déboguer l'algorithme d'étiquetage séquentiel avec correspondance entre points. Cette fonction fonctionne bien mais présente un bogue au bout de quelques temps. De plus la détection peut être améliorée pour détecter des objets peu lumineux et des objets de couleurs. Pour finir, l'algorithme de suivi de mouvement doit être fini pour traiter tous les cas possibles. De multiples solutions existent, d'autres approches peuvent être développées. Par exemple travailler dans les infra-rouge à l'air plutôt prometteur !

## **9 CONCLUSION**

La première version de l'algorithme fonctionne de la capatation à la communication. Ce programme peut être utilisé pour des cas où la fiabilité de suivi de mouvement n'est pas nécessaire.

Les résultats trouvés sont bien entendu différents de ceux escomptés. Tous les objectifs fixés n'ont pu être tenus, faute d'expérience et de temps. J'ai eu du mal à évaluer le temps de développement de différentes tâches.

Cependant, je suis resté motivé et confiant jusqu'au bout. D'avantage de temps m'aurait permis de m'y consacrer d'avantage et de satisfaire tous les objectifs fixés.

J'ai beaucoup appris techniquement et enrichi ma connaissance de par cette expérience supplémentaire et par les lectures de documentations techniques et l'utilisation d'OpenCV.

Je me suis surtout rendu compte de la difficulté à automatiser des tâches qui sont très simples pour un humain mais prennent une complexité conséquente à la recréer.

## 10 REFERENCES

processing : <http://processing.org/>

oscpack : <http://www.audiomulch.com/~rossb/code/oscpack/>

Open Source, Computer Vision Library, “Reference Manual”OpenCV :  
<http://www.intel.com/technology/computing/opencv/>

“Numérisation et traitement d’une partie d’échecs avec une caméra” de ABDELWAHAB LAHLOU Amine, FROGER Arnaud, SIMON Florent

“Projet de télédétection” de Germain Barret, Mourad Benoussaad, Estelle Brémont Gergana Gocheva, Emeric Golfier, Guillaume Lambert, Yan Zhang

“Pure Data : network and Applications ” de SM 1205 Interactivity

“ The Design and Evolution of Fiducials for the reacTIVision System” de Ross Bencina and Martin Kaltenbrunner, Music Technology Group, Audiovisual Institute Universitat Pompeu Fabra, Barcelona

Analyse d'images : “ filtrage et segmentation ” de [Cocquerez](#)

## 11 ANNEXES

### 11.1 Utilisation de la bibliothèque OPENCV sous DevC++

OpenCV is Intel's open source computer vision library. It contains a huge collection of computer-vision related algorithms, data structures, filters, etc. that we may find useful.

Once your IDE is set up, go to <http://sourceforge.net/projects/opencvlibrary/> and click on the opencv-win link

Click on the Download OpenCV Beta5 link. On the next page, choose a mirror to download

Run the setup program you just downloaded. Accept all the defaults; this will install OpenCV to C:\Program Files\OpenCV.

If you navigate to the C:\Program Files\OpenCV\samples\c directory, you will find a bunch of C/C++ code samples, showing off some of OpenCV's capabilities. Some of them require a connected webcam to work; those won't do anything if you open them up. You can see some examples of edge detection, erode/dilation, etc.

**IMPORTANT:** There is a naming problem with the latest beta5 version of openCV. Navigate to the C:\Program Files\OpenCV\bin directory. There you will find several .dll files with an 097 in them. Copy each .dll, and change the name from 097 to 096. (i.e., the copy of cv097.dll becomes cv096.dll, same for cvaux, cvcam, cxcore, and highgui) (this can be done by running `copy *097.dll *096.dll` under the command shell)

Add the C:\Program Files\OpenCV\bin directory and the C:\Program Files\OpenCV\otherlibs\highgui directory to your system PATH variable. This tells the computer where to find the .dll system libraries. To do this, go to Start Menu->Control Panel -> System (or hit the Windows Button + break keys). Go to the Advanced tab and click the Environmental Variables button. In the lower box, find the variable labeled PATH. Click Edit. In the lower text box, navigate to the end and append ;C:\Program Files\OpenCV\bin;C:\Program Files\OpenCV\otherlibs\highgui (remember the semicolons). Click OK in all the windows.

Open up Bloodshed Dev-C++

Go to File->Open Project or File and open up C:\Program Files\OpenCV\samples\c\contours.c

Make a new file (File->New->Source File). Copy and paste the code from contours.c into this new file, and save it as test.cpp

Go to Tools->Compiler Options. It should come up with the Compiler tab selected. Check the box labeled **Add these commands to the**

**linker command line and type (or copy-paste) the following into the text box:**

```
-lhighgui -lcxcore -lcvcam -lcvaux -lcv
```

This tells the IDE to link your program with the OpenCV libraries.

Switch to the Directories tab, and in the Libraries sub-tab add to directories:

```
C:\Program Files\OpenCV\lib
```

Switch to the C includes tab. Add

```
C:\Program Files\OpenCV\cxcore\include
C:\Program Files\OpenCV\cv\include
C:\Program Files\OpenCV\otherlibs\highgui
C:\Program Files\OpenCV\cvaux\include
C:\Program Files\OpenCV\otherlibs\cvcam\include as include directories.
```

Do the same thing for the C++ includes tab.

Now you should be able to compile and run the test.cpp sample file with the command Execute->Compile and Run. This program analyzes the image of the faces to extract information about the different regions. You can change the level of the contour to see different elements of the face (eyes, pupil, nose, mouth, etc). Play around with the source code for this program; change parameters or a line of code, and see what it does. Make sure to recompile with (Execute->Compile and Run) every time you make a change.

NOTE: You might receive an error "Application failed to start because cv097.dll was not .....". To resolve this issue you need to copy the following three .dll files from the directory C:\Program Files\OpenCV\bin to the directory C:\Program Files\OpenCV\samples\c

- 1.cv097.dll
- 2.cxcore097.dll
- 3.highgui097.dll

## 11.2 Utilisation de la bibliothèque OPENCV sous VisualC++ 6

- **Installation.**

- Installer la librairie OPENCV disponible sur la page suivante :

- [\(http://sourceforge.net/projects/opencvlibrary/\)](http://sourceforge.net/projects/opencvlibrary/)

- Créer un projet Win32ConsoleApplication.

*Les chemins suivants permettant de lier la bibliothèque, sont des chemins absolus considérant l'installation d'OPENCV dans le répertoire par défaut : C:\Program Files\OpenCV\*

- Dans le menu "Project", "Settings", "C/C++", category "Preprocessor".

Ajouter les liens suivants dans "Additional include directories":

**C:\Program Files\OpenCV\cv\include, C:\Program Files\OpenCV\cvaux\include, C:\Program Files\OpenCV\cxcore\include, C:\Program Files\OpenCV\otherlibs\highgui**

- Dans le menu "Project", "Settings", "Link", category "Input".

Ajouter les librairies suivantes dans "Object/library modules": **cv.lib cvaux.lib cxcore.lib highgui.lib**

Ajouter les liens suivants dans "Additional library path": **C:\Program Files\OpenCV\lib**

- Modifier la variable d'environnement "PATH" de Windows en ajoutant: **C:\Program Files\OpenCV\bin**

### **11.3 EXEMPLE DE CREATION INTERACTIVE MULTIMEDIA : Corps, graphisme et son**

Exemple sur <http://fredalemany.zeblog.com/c-general>

#### **Installation / Spectacle**

Cette installation permet au public de générer une composition sonore et une composition graphique par son déplacement dans l'espace et par son contact avec d'autres personnes. Cette composition va varier en fonction du nombre de personnes présentes, de leurs déplacements et de leurs regroupements éventuels. A un moment programmé, des danseurs se mêleront au public pour jouer du même dispositif de génération sonore et visuelle. Il s'agit de mettre en situation d'interaction, le public et les artistes, avec un dispositif de diffusion musicale et graphique.

#### **Captation des déplacements corporels**

Le dispositif devra permettre dans un espace donné, 8 X 6 mètres, d'identifier et de repérer le déplacement de chaque personne présente, public et acteur. Deux caméras vidéos situées au dessus de l'espace sensible fourniront après l'analyse et la reconnaissance de formes la position à tout instant de toutes les personnes entrant dans cet espace. L'ensemble du dispositif sera sensible à la saturation de l'espace réel, sonore et visuel. Trop d'individus dans un même espace produisent ils nécessairement du bruit et de la fureur ?

#### **Champ de réflexions**

Dans la mise à jour du mécanisme cyclique de la violence fondatrice, la science de l'homme et de la culture trouve un principe simple et terrible. Comment se soustraire à ce mécanisme archaïque et comment élaborer de nouveaux modes de mise en relations des groupes sociaux ? S'y soustraire pour que les collectivités n'aient plus recours à la violence pour reconstituer leurs identités sociales. Elaborer de nouveaux modèles culturels pour maintenir un être social et collectif en cohésion. La création artistique participe à cette réflexion et c'est dans ce champ qu'elle invente de nouveaux rituels ludiques et participatifs.

#### **La composition chorégraphique : la danse contact improvisée**

Huit acteurs et danseurs seront invisibles au milieu du public. Ils entreront progressivement et individuellement en jeu avec le dispositif en explorant plus précisément ses possibilités. Puis ils provoqueront des rencontres corporels avec le public. La composition chorégraphique est d'abord présente pour mettre en situation le dispositif et le pousser plus loin que celle entamée par le public au hasard du jeu. Les danseurs auront eu le loisir de jouer et de répéter des séquences de déplacement qui produiront des sonorités et des images étudiées. La chorégraphie sera composée pour utiliser au mieux le dispositif. En retour le dispositif sera conçu pour accompagner au mieux la composition chorégraphique.

#### **Génération des sons**

La composition sonore doit permettre à chacun d'entendre la variation sonore que produit son corps en fonction de son déplacement, de sa position dans l'espace mais surtout de sa position par rapport aux autres personnes. La composition musicale doit amplifier les rencontres corporelles dans cet espace et provoquer la dissonance, la confusion aussi bien que l'harmonie et si possible une pulsion rythmique. L'essentiel de la musique matérialise les distances et les interactions entre les personnes dans cet espace.

## **Génération des images**

La composition graphique se fera en fonction des informations fournies par le dispositif et tiendra compte de la position, de la vitesse, du parcours et des contacts entre les personnes. Chaque personne produira une trace graphique sur les écrans de projections d'images. La composition graphique suivra le déplacement de chaque personne qu'elle projettera sur une surface à deux dimensions, le plan de l'écran. L'écran principal doit traduire la situation vue de dessus de l'espace du dispositif. Chaque personne est un pinceau d'une couleur spécifique et qui marque une trace sur l'écran, plus ou moins forte, plus ou moins durable et qui varie en fonction de la vitesse. Chaque aura peut être animée d'un mouvement propre, nuage de point, mouvement de matière. Les interactions entre les personnes peuvent produire des phénomènes graphiques plus autonomes suivant des modèles qui peuvent être météorologiques, physiques, d'attractions ou de répulsions.

## **Dispositif d'interaction**

Le dispositif est une installation techniquement autonome constituée de deux caméras vidéos, de deux unités centrales, de 2 systèmes d'amplifications sonores, de deux vidéos projecteurs, de 4 enceintes et de deux écrans de 4 x 3 mètres. La scénographie est construite par deux murs identiques placés face à face et séparés par l'espace interactif. Chaque mur sert d'écran de projection, de pont lumière, d'accroche pour le système sonore, d'abri pour la partie technique, d'élévation pour la caméra vidéo et de projecteur d'images pour le mur situé en face. L'ensemble est un volume de 4 mètres de hauteur, 4 mètres de largeur et 8 mètres de longueur qui peut s'ouvrir et se fermer avec des pendrions suivant les conditions de luminosité.

## **La réalisation de la maquette**

Cedric Doutrillaux, Luc Martinez, Jean Noël Montagné, Etienne Rey, Jean Marc Matos, Antoine Schmitt, Nicolas Clauss sont invitées au Hublot pour présenter leurs œuvres au public et échanger avec d'autres artistes en coordination avec la programmation de l'espace culture multimédia. Au cours de l'année 2006 un ensemble de rencontres et d'étapes de conception et de réalisation sont prévus avec des programmeurs, des musiciens, des graphistes et des danseurs pour constituer une équipe de travail. Plus spécifiquement il est prévu des ateliers de danse contact, des ateliers d'improvisations musicales avec des compositeurs électroniques, des ateliers entre programmeurs, musiciens et graphistes, des installations, des performances publiques. L'équipe de travail se constituera pendant ces rencontres et comprendra une chorégraphe, probablement Emmanuelle Pépin de la Cie vertige, un programmeur en C++ probablement Edouard Galera, un programmeur Pure data, un compositeur sonore, un graphiste plasticien et des techniciens. La maquette de présentation rendra compte d'une simulation clavier/souris du dispositif sur Pure data.

## **La production**

Au début de l'année 2007, dans un lieu de résidence artistique probablement l'Entre-Pont de la Halle spada, dans un temps assez court de deux mois la création se finalisera. L'ensemble du dispositif sera déjà fonctionnel. Les éléments sonores et graphiques seront constitué mais non composé. L'ensemble de la composition se fixera par des répétitions, des essais et aboutira par une écriture musicale, chorégraphique et picturale parfois traditionnelle. La première pourrait avoir lieu au printemps 2007.



## **La diffusion**

Il faut imaginer que le dispositif s'installe dans à peu près n'importe quel type d'espace, entrepôt, place publique, salle polyvalente qui peut comprendre une surface d'interactivité de 8 X 6. Il s'installe aussi bien dans des espaces pluridisciplinaires, dans des festivals de rue, dans des lieux d'expositions, dans des espaces de circulation du public, dans le hall des théâtres.

## **Les lieux et festivals dans les Alpes Maritimes**

Halle Spada à Nice

Festival de théâtre de rue des Siacrerries à Carros, directrice Sylvie guigo

Le pré des Arts à Valbonne, directeur Olivier Colladant

Forum danse de Monaco

Festival de danse de Cannes

Théâtre National de Nice, directeur Daniel Benoin

Théâtre de Grasse, directeur Jean Florés

## **Conception du projet : Frédéric Alemany**

Né le 17 septembre 1966 à Nice / Footballeur à l'U.S. de Cagnes sur mer de 1974 à 1984 / Etudiant de 1970 à 1990 / Licence de philosophie à l'Université de Nice de 1984 à 1988 / BTS électronique numérique en formation professionnelle de 1998 à 1990 /

**Technicien informatique**, employé par la Société ReadKey , Bureau d'études électroniques à Nice sous la direction d'Edouard Galera / Gestion informatique de l'association Niss Art Production sous la direction de Cathy Ruf / Ouverture d'un espace culture multimédia / Ouverture du hublot, centre de création multimédia / 1 ère Création multimédia interactive, le Robot Erectus /

**Comédien et danseur**, Performances individuels / Comédien de rue dans deux créations, « Le charivari », « Otez l'eau de Venise » sous la direction de Serge Dotti / Participation au Carnaval indépendant de St Roch avec Nux Vomica. / Comédien de théâtre avec la Cie la Saeta avec Marie Jeanne Laurent dans deux créations, Quai Ouest de Koltés, No Moderne de Mishima / Comédien danseur avec la Cie Etat de rue sous la direction de Patrick Tridon. et Joelle Donati . Danse en rue. Roméo et Juliette / Comédien danseur avec la Cie Jabiru sous la direction de Pascale Flores / Chanteur et auteur du groupe de rock musette des Afro Baleti.avec Johanna Piraino / Fondateur et musicien de la fanfare des diables bleus / Comédien danseur avec la Cie Antipode sous la direction de Lysie Philip /

**Chargé de production**, Fondateur de l'association Diva production spectacle vivant / Fondateur de la Cie Ram-Dam, Théâtre de rue à Nice / Fondateur du collectif des diables bleus, lieu de création artistique alternatif / Organisateur d'Octobre Bleu / Fondateur de l'Entre-Pont, lieu de création spectacle vivant et nouvelles technologies / Aménagement de la Halle Spada à Nice /

**Metteur en scène**, 1 ère Création, spectacle de rue et danse : La Cité Plastique créée au Carnaval de l'Ariane représentée aux Siacrerries de Carros et à la quinzaine des Compagnies Paca au Théâtre de Nice /

## Première page du projet

Par Bozaris :: 13/11/2005 à 11:42

Ce blog a pour but de faciliter la coréalisation du projet. Les billets (articles) ne seront pas profondément modifiés pour maintenir la trace de l'évolution de la conception du projet.

Je vais décrire dans cette première page les orientations de ce projet.

Il s'agit de mettre en situation d'interaction le public et les acteurs (danseurs) avec un dispositif de diffusion sonore et de projections d'images.

### **1/ Dispositif d'interaction**

Le dispositif devra permettre dans un espace donné, une salle de 1000 m<sup>2</sup>, d'identifier et de repérer le déplacement de chaque personne présente, public et acteur. Le dispositif est une installation autonome.

### **2/ Interaction public-crédation sonore**

La programmation musicale se fera en fonction des informations fournies par le dispositif et tiendra compte de la position, de la vitesse, du parcours et des contacts entre les personnes. La composition variera en fonction de la composition des interactions corporelles. Elle sera transmise en multidiffusion.

### **3/ Interaction public-crédation graphique**

La programmation graphique se fera en fonction des informations fournies par le dispositif et tiendra compte de la position, de la vitesse, du parcours et des contacts entre les personnes. La composition graphique suivra le déplacement des personnes qu'elle interprétera sur une surface à deux dimensions (un plan de l'espace).

Elle sera transmise par deux vidéoprojecteurs sur deux écrans.

### **4/ La création danseur-dispositif**

La chorégraphie sera composée pour utiliser au mieux le dispositif. Les acteurs seront invisibles au milieu du public et provoqueront des contacts corporels. Peu à peu la gestuelle se définira entre les danseurs et évoluera vers une forme traditionnelle de représentations.

## 11.4 Chromakey

Cette « chromakey » a pour but d'extraire un objet en couleur. Voir image ci-dessous.

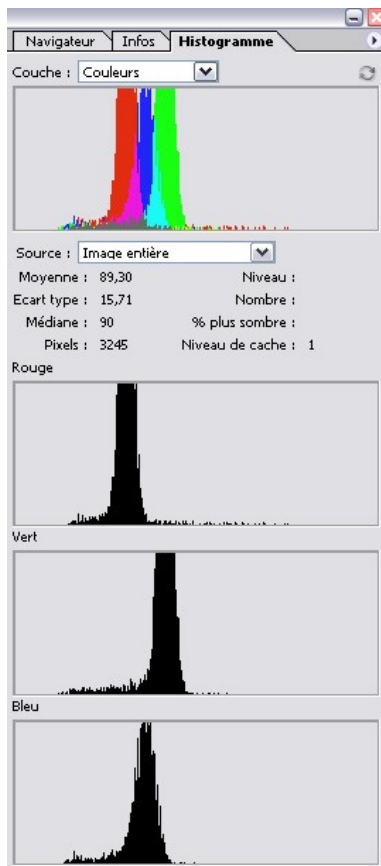


*Illustration 16: Billard*



*Illustration 17: Boule de billard rouge*

Tout d'abord, la partie sélectionnée est étudiée avec son histogramme et ses caractéristiques.



*Illustration 18:  
Histogramme de la partie  
sélectionnée*



*Illustration 19: Résultat  
après une "chomakey"*

L'histogramme sur la zone sélectionnée ne permet pas de d'extraire la boule. Par contre, l'histogramme de la composante verte est plus éloigné des autres histogrammes. L'illustration 19 est le résultat d'une extraction de fond.