An Odometry-Based Approach for Indoor Object Viewpoint Recognition

Luigi F. Tedesco[1], Céline Craye[2], Jean-François Goudou[3] and David Filliat[4]

Abstract— **Object recognition capability is a essential condition for giving autonomy to mobiles robots in human made environment. However, achieving this goal by means of visually representing objects is a hard task ?? and using all possible sources of information is a must. Here we present a procedure to incorporate the notion of continuity and overcome ambiguous points of view. By observing objects from different perspectives bound with a Markovian modeling of the stochastic processes of recognizing each of the objects viewpoint, the algorithm copes with a sparse database, blurred images from motion and object spatial symmetry, to recognize and estimate objects 6-dof pose. A multi-modal Kalman based tracking was also implemented in order to recognize multiple objects simultaneously. The approach was tested in a mobile platform and the comparison between the single viewed and the proposed recognition gave promising results, reinforcing the proposal of blending odometry and recognition.**

I. INTRODUCTION

The vast majority of the literature focus on single image object visual recognition for helping robots in tasks such as semantic navigation **??**, pose estimation for grasping **??** and environmental search **??**. Typically, a set of features is extracted from a segmented object candidate and, subsequently, compared to a database of priori known objects. Extensive work have been done in order to increase efficiency in each one of the sub-processing steps. Among them : segmentations methods using range cameras, features that describe color and texture **??**, geometry **??**, contours **??**, besides classifiers and matching techniques. Alternatively, a deep neural architecture **??** can perform a direct object visual classification after a delicate training phase. However, the classic recognition pipeline seems to be more natural and simple to be implemented with a straight-forward training, still having reasonable results.

Nevertheless, ambiguous viewpoints easily trick visual descriptors reducing its recognition capability. Observing objects sequentially from distinctive points of view seems to be a natural way to deal with the problem. A solution inspired by human behavior for learning new unseen objects has been proposed by **??**, using key-frames and the rate of matching features with past frames, to overcome ambiguity in face recognition task. More work have been done to model objects different viewpoints perspectives summarized by Roy and al. **??**.

[1]Luigi Franco Tedesco is an undergraduate student with Faculty of Electrical Engineering, Robotics and Artificial Intelligence, Thales Service, 91767 Palaiseau, France tedesco@ensta.fr

[2]Céline Craye is with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA b.d.researcher@ieee.org

[3]Jean-François Goudou is with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA b.d.researcher@ieee.org

[4]David Filliat is with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA b.d.researcher@ieee.org

II. PROPOSED APPROACH

A. General archtecture

Our approach was designed for robots equipped with odometric and RGB-D sensors. Informations from these units are sent to a processing module that isolates objects from input images, extracts their characteristics through features, and compare them with a reference database stored in memory. When no matching is found between observation and the current database, a new object can be added to the later, increasing the robot's knowledge about the environment.

The architecture of the system is illustrated in Figure 1 and explains the dependencies between the different processing modules and the information flow through them.

More precisely, the processing units takes as input the point cloud from the RGB-D sensor as well as the odometry measure of the two wheels. The first module consists in a segmentation step that cleans up the point cloud by isolating objects of the scene into clusters of points. Those clusters are then sent to the feature extraction module to convert them to discriminative description histograms. Simultaneously, a cartography module converts the odometry data and the segmented object image position into the localization of scene elements in the world referential. Finally, the recognition module uses both the object feature and the evolution of those features compared to the displacement of the robot to retrieve the most likely object and view.

B. Object Segmentation

The segmentation step aims to differentiate objects from the background of raw images. Stereoscopic and infra-red cameras helped the treatment adding a new dimension to images, allowing segmentation geometrically. In the case of motionless sensors, statical background subtraction approach are typically used for segmentaiton [5]. This is not applicable in our case as the robot is constantly moving in its environment. By making the hypothesis that objects are represented in the scene as cluster of points right above a main plan, approaches such as *Tabletop object detector* [6] and Caron et al. [4] determine the main plane's convex hull and search for objects clusters inside it. The later algorithm also removes plans orthogonal to the ground, considered as walls, and is more suitable for finding objects placed on ground level in indoor environments, therefore, explaining our choice for it. Furthermore, in our robot assemble, the

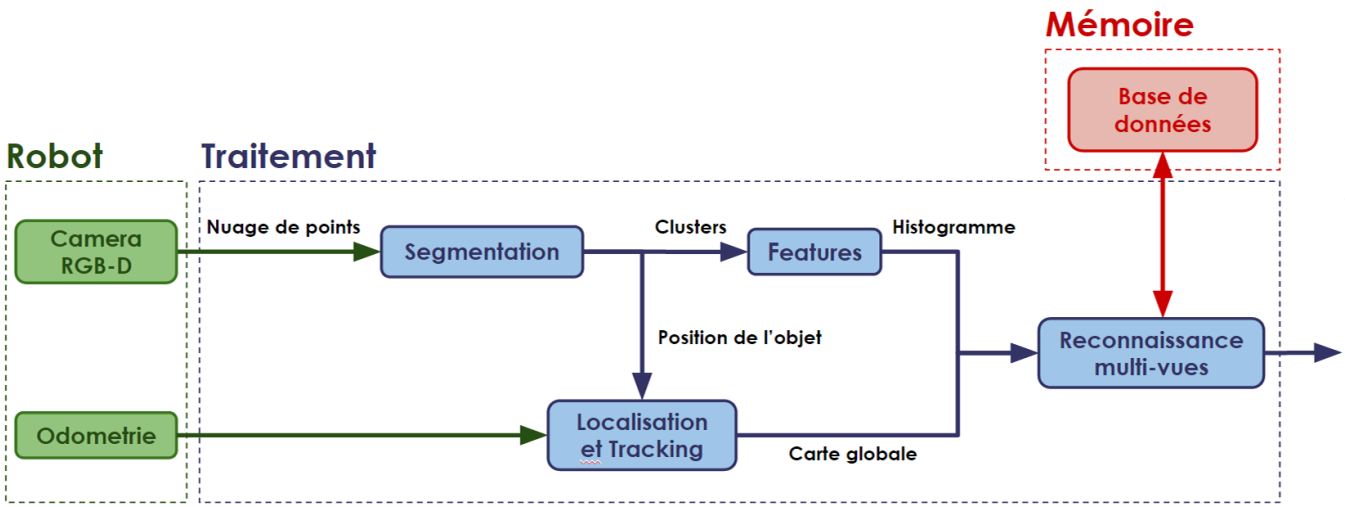

Fig. 1. General achitecture of the system

camera is always keeping the same orientation towards the floor, in this manner, its plan equation can be estimated once during a calibration step and used during the whole experiment.

C. Object representation

The definition of object used in segmentation is enough for separating them from the environment, however for a more complex classification task, its unique characteristics must be extracted. The first dimension of information relates to its position in the environment. This cartographic data allows the robot to track objects while moving around the scene, consequently associating multiple views of it. A second dimension concerns the objects' intrinsic characteristic capable of differentiating them from one another that can be translated in terms of visual and geometrical features.

1) Features: The concept of features tries to transpose those intrinsic characteristics from objects into a lower dimension space. At the same time, they are required to be stable under environmental changes and affine transformations for being truly representative. Several features were proposed by the literature to incorporate the most diverse characteristics. Among three dimensional geometrical ones one can state local descriptor SHOT [8] and semi-global CVFH [2], [3] descriptors. The Viewpoint Feature Histogram **??** - VFH - captures in a single histogram the object's geometry by estimating the angular transformation between the normal of each of the its points and the standpoint from where it has been viewed. The interest of using such geometrical features is to explore the ambiguity created from objects' spatial symmetry, even if color and texture descriptors could enhance recognition.

2) Aspect-graph: In order to represent objects in the three dimensional space, a model that captures both cartographic and intrinsic components is mandatory. Additionally, object

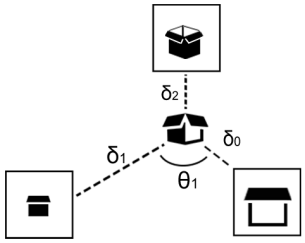

Fig. 2. Reprsentation des objets par un modle polaire

visual characteristics are viewpoint-dependent, thus, requiring an additional treatment that correlates appearance and the necessary movement to transit between them.

The above requirements were translated into a polar referential, illustrated in 2 where nodes represent the object aspect from a certain point of view and transitions from nodes can be geometrically deducted.

Usually, visual features like SIFT are scale-variant as the image resolution plays an important role. Viewpoint information can bias the classification to search for eligible candidates in the database.

D. Object Classification

The viewpoint recognition phase consist on finding the closest element in the database given a test feature to be matched. This classification problem could be solved by classic machine learning techniques. Nevertheless, the work from Aldoma et al. [1] suggest the use of a K-Nearest Neighbors with a chi-squared distance between histograms for classification. This classifier has the great advantage of having both classification and training stages extremely fast for the intended database size.

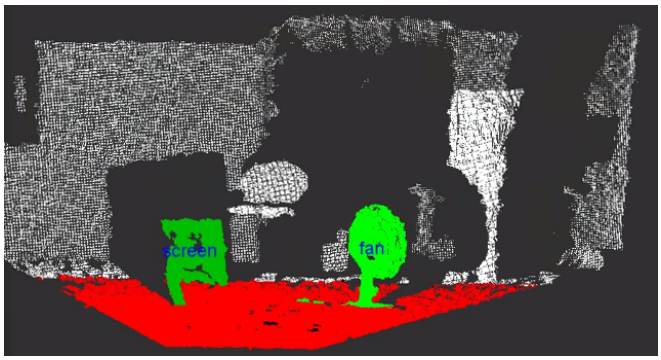

Fig. 3. **Single-image recognition** - Object identification for two segmented objects - a computer monitor and a small electric fan. The red points represent the floor plan and the ignored white ones exceed the three meters threshold. One could also notice the infra-red shadows created by objects that possible can occlude others

E. Multi-object Tracking

A tracking system is necessary to follow each of the objects on sight. Imprecisions from communication delays between processing modules, segmentation and object centroid calculation call for a multi-modal filtering method for estimating their absolute position. The proposed multi-tracker runs a Kalman filter for each object where each new observation either updates the nearest existing filter or creates a new one if the closest distance to the all filter's states is bigger than a threshold.

F. Reconnaissance Multi-vue

Moving the robot around the environment produce a series of object observations. The idea behind our system is to find consistency between this sequence of possibly misrecognized objects given a known displacement. In orther words, odometry information can reinforce some viewpoint candidates as they are consistent with the movements prediction... (???)

1) Hidden Markov Model: Considering that a single image and the robot movement is able to predict the next view of the object, ensuring the first order Markov property and allows the stochastic process viewpoint recognition to be modeled as a Markovian process. The probability of transition between views is calculated by the triangularization of the robots initial and final position and the

Le dplacement physique du robot produit une squence d'observations d'un mme objet, sous diffrents points de vues. On exploite l'information odomtrique entre les diffrentes vues pour renforcer l'estimation de la vue d'un objet. De cette manire, l'volution de la reconnaissance au cours du temps est reprsente par un processus stochastique, dont une modlisation possible consiste le traiter de faon discrte dans un espace d'tat. Ayant l'apriori que la dernire image et le dernier dplacement suffisent pour faire cette prdiction (c'est-dire en respectant la proprit de Markov de premier ordre), le processus stochastique est modlise par une chane de Markov cache.

Concrtement, les tats cachs correspondent des vues d'objets prsents dans la base de donnes et dj stocks dans la mmoire du robot. Cela contraint le nombre d'tats et garantie que la chane soit finie. Puis, une matrice de transition $a_{i,j}$ dcrit l'volution du processus. C'est cette matrice de transition qui permet de prendre en compte l'odomtrie et la transition entre les vues d'un mme objets. Enfin, une autre matrice, $\mathrm{P}(y_1 \mid k)$, dite matrice d'mission, estime la vraisemblance entre l'observation et les tats de la chane.

Plus prcisment, $a_{i,j}$ est dfinie en fonction de l'angle δ_{angle}, calcul par 1 qu'a parcouru le robot par rapport l'objet entre deux vues successives. Dans notre modle, on considre que $a_{i,j}$ est nulle si i et j sont deux vues d'objets diffrents. Pour deux vues i et j d'un mme objet, spares d'une distance d, le poids accord $a_{i,j}$ sera d'autant plus fort que δ_{angle} et d sont proches. D'autre part, la matrice d'mission $\mathrm{P}(y_1 \mid k)$ correspond la similarit entre l'histogramme d'un objet segment y et celui d'une vue d'objet dans la base de donnes k, normaliss par l'quation 2. La similarit est calcule comme l'inverse de la distance Chi square dfinie la section II-C.2.

$$\begin{aligned} \vec{d}_0 &= p_0 - p_{obj} \\ \vec{d}_1 &= p_1 - p_{obj} \\ \delta_{angle} &= atan(\vec{d}_1) - atan(\vec{d}_0) \end{aligned} \tag{1}$$

La transformation des distances des histogrammes en probabilit est faite d'aprs la normalisation suivant :

$$\mathbb{P}(y|x, database) = \frac{\sum_a d_a^x - d_y^x}{\sum_b \sum_c d_c^x - d_b^x} \tag{2}$$

o x est l'image de test et y un lment de la base de donnes. Dans le cas du plus proches voisin, la normalisation ne prend en compte que les k plus proches histogrammes, par opposition une approche *brute force*.

Une autre modlisation possible aurait t d'avoir une chane de Markov cache distincte pour chaque objet et ensuite dcider chaque pas de temps le processus le plus vraisemblable. Cette modlisation peut tre vue comme un sous-ensemble du cas prcdent o les transitions entre deux objets ne sont pas considres. Pourtant, il peut arriver que deux objets soit considrs comme positionns au mme endroit, ou bien que des objets mobiles fusionnent (par exemple, une personne qui viendrait s'asseoir sur une chaise, ou encore une personne qui commence marcher) [1].

2) Algorithme de Viterbi: Reste donc extraire des informations de la modlisation Markovienne propose. La squence d'tats la plus vraisemblable qui pourrait avoir gner les observations $y_1, \ldots, y_T$, correspond normalement la squence d'objets reconnus. Afin de retrouver cette squence, aussi appelle chemin, on fait appel la programmation dynamique, et plus spcifiquement l'algorithme de Viterbi, d'o le nom chemin de Viterbi. L'algorithme retrouve de faon rcursive l'tat courant le plus probable, en prenant en compte seulement les observations jusqu' un instant donn et son estimation aux instants antrieurs. Ceci se traduit par les quations 3

[1] Le fait de se mettre en mouvement altre les formes d'une personne, ce qui rend possible sa dtection comme un nouvel objet.

$$\begin{aligned} V_{1,k} &= \mathrm{P}\big(y_1 \mid k\big) \cdot \pi_k \\ V_{t,k} &= \max_{x \in S} \big(\mathrm{P}\big(y_t \mid k\big) \cdot a_{x,k} \cdot V_{t-1,x}\big) \end{aligned} \tag{3}$$

Ici, $V_{t,k}$ reprsente la probabilit que la squence d'tats la plus probable finisse dans l'tat k, ayant gnr les observation l'instant t, tandis que π_i reprsente la probabilit initiale de se retrouver en chaque tat. Pour retrouver le chemin de Viterbi, il suffit de trouver le maximum de $V_{t,k}$:

$$x_T = \arg\max_{x \in S}(V_{T,x}) \tag{4}$$

III. Experimental results

The proposed recognition system was deployed in a differential mobile robot, Wifibot V2, embedded with a RGB-D camera, Asus Xtion Pro Live. The algorithm architecture were implemented over ROS using PCL and OpenNi2 libraries.

In the interest of validating the approach, the robot was initially taught aspects graphs from chosen objects and two sets of experiments were proposed to analyze the efficiency of the algorithm in real scenarios.

A. Object Database

First, twenty objects varying in size and form (monitor, boxes, chairs, trashcans, people, ...) were selected to compose the knowledge database of the robot. The aspect graphs of each object weas composed by VFH features from *eight* equally distant viewpoints acquired from positioning the robot around the to be learn object 1.5 meters away.

B. Odometry Contribution

The first experiment consist in a performance comparison between the single image and odometry-base recognition techniques. In other words, this comparison attest whether the proposed architecture is interesting or not, since having the same or worst performance by the cost of adding a complex post-processing and tracking modules is not interesting.

Technically, each object was partially circled by the robot from at least four random initial positions at $0.35 \pm 0.1 m/s$. The robot recorded for each run between 5 and 30 frames at different viewpoints of the object, perhaps inexistent in the database, and try to recognize the object and estimate its pose using both recognition algorithms.

C. Multi-object recognition

In a second evaluation, concurrently with the recognition efficiency, the multi-tracking capabilities were put into test.

La difficult de l'valuation vient, premirement, du fait que la base de donne a t ralise avec trs peu de vues, ce qui donne lieu de mauvaises reconnaissance mono-vue lorsque le point de vue observ se situe entre deux vues de la base de donnes. De plus, la vitesse de dplacement peut gnrer des images plus floues lors des acquisitions et la modification de l'angle de la camra[2] apportent un obstacle en plus pour le *matching* de descripteurs dans le classificateur K-NN.

[2]L'angle entre la base et la tte de la camra Asus Xtion est facilement modifi.

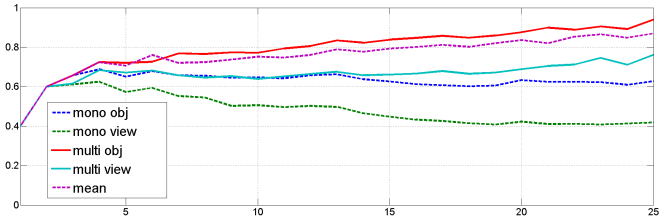

Fig. 4. **Rsultat de l'valuation** - Les courbes en pointills reprsentent la reconnaissance base sur une seule image (mono-vue), tandis que les courbes pleines correspondent au systme multi-vues. Avec un nombre rduit d'observations, la reconnaissance mono-vue tend tre lgrement plus performante. L'cart se creuse mesure que ne nombre d'observations augmentent, jusqu' un cart de 33 % une fois le tour complet de l'objet effectu, soit 92% de russite pour l'estimation de l'objet et 74 % pour l'estimation de l'orientation.

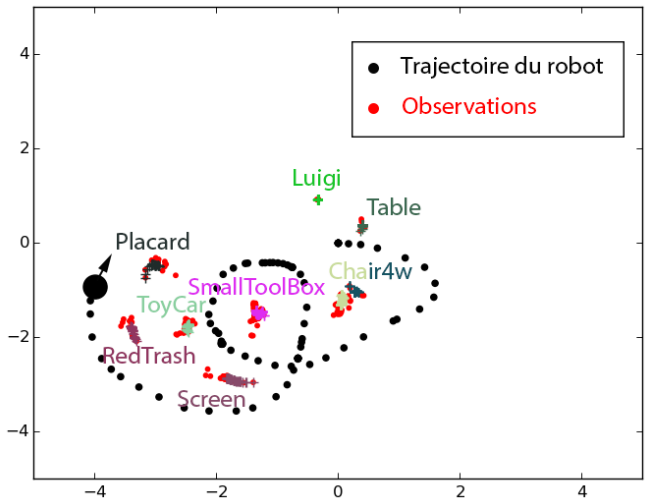

Fig. 6. **Rsultat du suivi multi-cibles** - Les points rouges correspondent aux centrodes des objets segments. Chaque croix de couleur reprsente une estimation de position d'un filtre de Kalman, avec une couleur par filtre. En thorie, un filtre devrait tre associ chaque objet. En noir la trajectoire du robot au cours du temps.

Un exprience typique est illustre dans l'image 5 :

La premire ligne correspond la squence d'images vues par le robot chaque instant de temps, et donc, l'objet tre reconnu. La seconde ligne, donne par l'algorithme de reconnaissance, quivaut la vue la plus probable de l'objet reconnue par le K-plus proches voisins. Il est intressant remarquer que l'invariance rotation du descripteur trompe l'estimation de l'orientation en prenant son correspond nantiomorphe dans le premier carr rouge. Autrement, le dos du pinguin tant une grande surface presque plane, il est partiellement retir par l'tape de segmentation. Ainsi, le nuage de points rsultant de ce point de vue n'est pas suffisamment complet pour caractriser correctement l'objet, ce qui induit une mauvaise reconnaissance dans le carr bleu. Au final, on remarque que le traitement apport par la chane de Markov cache permet de corriger les problmes d'une base de donne relativement sparse avec des possibles erreus de segmentation, permettant la correction simultane de la reconnaissance de l'objet et de son orientation en ligne.

Le rsultat de l'valuation peut tre reprsent sous forme

Fig. 5. **Expriment typique** - Reconnaissance multi-vue corrige des ambiguts malgr la mauvaise segmentation lors de la cration de la base.

d'une courbe : le nombre d'observations pour un mme objet en abscisse, par rapport aux pourcentage de reconnaissance d'objet et de vue, pour les reconnaissance mono et multi-vues.

La courbe de la figure 4 permet de conclure, tout d'abord, que l'algorithme de reconnaissance multi-vues est plus performant que sa correspondante mono-vue lorsque le nombre d'observations augmente suffisamment.[3]. Dans un deuxime temps, on constate que l'estimation de l'orientation de l'objet est plus difficile estimer que sa reconnaissance, que cela soit pour la reconnaissance mono-vue ou multi-vues.

D. Suivi et reconnaissance multi-cibles

La deuxime exprimentent consiste placer des objets prsents dans la base de donnes dans une pice et conduire le robot en faisant en sorte qu'il les regarde sous plusieurs points de vues diffrents. Ce scnario est beaucoup plus complexe que celui d'avant. D'abord les objets s'occultent les uns les autres, donnant lieux de mauvaises segmentations. Ensuite, le suivi des objets est beaucoup plus complexe, car des objets proches peuvent tre confondus.

La carte finale donne par l'algorithme et localisant le robot ainsi que les objets dans le repre absolu est reprsente la figure 6. Une photo de la pice aves les objets disposs comme dans l'exprience est affiche en **??**.

Les rsultats de l'exprience montrent que le systme fonctionne encore, mme dans des cas beaucoup plus complexes. On notera toutefois une chute des taux de reconnaissance. Quelques amliorations proposes dans la section suivante pourraient tre utiles pour rendre le systme plus robuste:

REFERENCES

[1] A. Aldoma, Zoltan-Csaba Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R.B. Rusu, S. Gedikli, and M. Vincze. Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation. *Robotics Automation Magazine, IEEE*, 19(3):80–91, Sept 2012.

[2] Aitor Aldoma, Federico Tombari, Radu Bogdan Rusu, and Markus Vincze. *OUR-CVFH–Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation*. Springer, 2012.

[3] Aitor Aldoma, Markus Vincze, Nico Blodow, David Gossow, Suat Gedikli, Radu Bogdan Rusu, and Gary Bradski. Cad-model recognition and 6dof pose estimation using 3d cues. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 585–592. IEEE, 2011.

[4] Louis-Charles Caron, David Filliat, and Alexander Gepperth. Neural network fusion of color, depth and location for object instance recognition on a mobile robot. In *Computer Vision-ECCV 2014 Workshops*, pages 791–805. Springer, 2014.

[5] Ke-xue DAI, Guo-hui LI, Dan Tu, and Jian YUAN. Prospects and current studies on background subtraction techniques for moving objects detection from surveillance video. *Journal of Image and Graphics*, 11(7):919–927, 2006.

[6] Ros documentation. Tabletop object detector.

[7] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.

[8] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *Computer Vision–ECCV 2010*, pages 356–369. Springer, 2010.

[3]Le fait que la reconnaissance mono-vue chute avec le nombre de vues peut tre lie au fait que pour les premires secondes de l'exprience, le robot est immobile et correctement plac. Lors du dplacement du robot, les prises de vues sont globalement moins bonnes et font chuter le taux de reconnaissance.

Fig. 7. **Reconnaissance Multi-cible** - Quatre des cinq objets prsents dans la scne ont t correctement reconnus avec une estimation d'orientation raisonnable. Le premire objet (une personne) tait trop prs du ventilateur, les deux ont donc t confondus dans le multi-tracking.