

## Extraction des pièces d'un plan d'habitation

Hervé Locteau, Sébastien Macé, Salvatore Tabbone, Ernest Valveny

► **To cite this version:**

Hervé Locteau, Sébastien Macé, Salvatore Tabbone, Ernest Valveny. Extraction des pièces d'un plan d'habitation. Jean-Yves Ramel. Colloque International Francophone sur l'Écrit et le Document - CIFED 2010, Mar 2010, Sousse, Tunisie. pp.1-12, 2010. <inria-00493386>

**HAL Id: inria-00493386**

**<https://hal.inria.fr/inria-00493386>**

Submitted on 18 Jun 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# Extraction des pièces d'un plan d'habitation

Hervé Locteau\* — Sébastien Macé\*\* — Salvatore Tabbone\* — Ernest Valveny\*\*

Université Nancy 2 – Loria UMR 7503  
615, rue du jardin botanique – B.P. 101  
54602 Villers-lès-Nancy Cedex, France  
{locteau,tabbone}@loria.fr

Centre de Visió per Computador  
Edifici O, Campus UAB, Bellaterra (Cerdanyola)  
08193 Barcelona, Spain  
{sebastien,ernest}@cvc.uab.cat

---

**RÉSUMÉ.** Dans cet article, une méthodologie d'extraction des pièces d'une image de plan architectural est décrite. Nous présentons tout d'abord un algorithme de détection des lignes de l'image. Ces lignes sont analysées afin d'identifier les murs constituant le plan. L'extraction des pièces peut ensuite être apparentée à un problème de segmentation pour lequel chaque région correspond à une pièce. Dans ce contexte, le processus de résolution consiste à décomposer récursivement l'image jusqu'à l'obtention de régions quasi convexes. De même que la notion de convexité est difficilement quantifiable, le choix d'un découpage est également approximatif. Dès lors, nous prenons en compte des connaissances associées aux plans architecturaux, afin notamment d'obtenir un ensemble de pièces majoritairement rectangulaires. De premières expérimentations sur un corpus de documents réels illustrent la pertinence de l'approche.

**ABSTRACT.** In this article, a method to extract the rooms of an architectural floor plan image is described. We first present a line detection algorithm to extract long lines in the image. Those lines are analyzed to identify the existing walls. From this point, room extraction can be seen as a classical segmentation task for which each region corresponds to a room. The chosen resolution strategy consists in recursively decomposing the image until getting nearly convex regions. The notion of convexity is difficult to quantify, and the selection of separation lines can also be rough. Thus, we take advantage of knowledge associated to architectural floor plans in order to obtain mainly rectangular rooms. Preliminary tests on a set of real documents show promising results.

**MOTS-CLÉS :** Plans Architecturaux, Analyse d'Images de Documents, Rétroconversion, Détection de Lignes, Décomposition Approximativement Convexe

**KEYWORDS:** Architectural Floor Plan, Document Image Analysis, Reengineering, Line Detection, Approximate Convex Decomposition.

---

## 1. Introduction

Ces dernières années, il y a eu un essor considérable de logiciels d'ameublement ou de décoration d'habitations. Leur principe est de permettre à un utilisateur de virtuellement visualiser l'impact des modifications qu'il envisage pour son logement (achat de meubles, changement de tapisserie, etc.). Un problème rencontré dans ces logiciels est la nécessité de retranscrire de manière précise la structure de l'habitation considérée. Celle-ci peut être obtenue à l'aide de systèmes de dessin de plans qui peuvent néanmoins s'avérer être difficiles à prendre en main. Le but de nos travaux, qui s'inscrivent dans le cadre du projet européen ScanPlan, est d'offrir la possibilité de numériser un plan et de l'interpréter automatiquement. Plus spécifiquement, notre objectif est de détecter automatiquement les différentes pièces d'une image de plan d'habitation.

L'interprétation de plans d'habitations est un problème qui est loin d'être résolu, principalement en raison de la variabilité des notations qui sont utilisées dans ces documents (Lladós *et al.*, 1997, Dosch *et al.*, 2000). Il n'existe pas de norme d'un pays à un autre, les conventions graphiques pouvant même varier d'un architecte à un autre dans un même pays. Cela se traduit à la fois par la nature des informations présentes (indications de cotations, de surfaces, de noms de pièces, de projection de toiture, etc.) et par la manière de les représenter (un mur est parfois représenté par un trait épais, deux traits fins parallèles, un trait fin et un trait épais parallèles, etc.). À long terme, le but de nos travaux est de proposer un système d'interprétation à la fois *générique*, i.e. capable de traiter autant de notations que possibles, mais également *complet*, i.e. de l'image source jusqu'au plan correct. Tout en gardant au maximum une approche générique (nous nous sommes efforcés de faire abstraction des notations), nous avons choisi d'amorcer notre prototype en nous focalisant sur un sous-ensemble de conventions graphiques. La figure 1 présente deux exemples de tels plans. Malgré cette spécificité, nous nous démarquons de la majorité des approches existantes en visant un système complet. Seul le processus de correction des erreurs, dans lequel l'utilisateur sera impliqué, n'est pas encore implémenté.

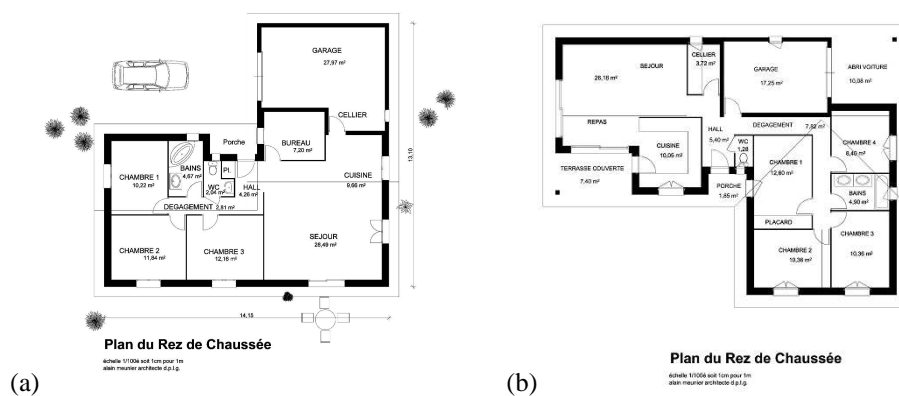


Figure 1. Deux exemples de plans d'habitation.

Dans ce papier, nous présentons les premiers résultats des recherches réalisées pour le projet ScanPlan. Tout d'abord, nous nous focalisons sur la détection des primitives graphiques qui nous intéressent, les lignes. Dans ce but, nous présentons une méthode originale, basée sur la transformée de Hough mais limitant un certain nombre de ses inconvénients en l'appliquant non pas sur l'image mais sur le résultat de la vectorisation de celle-ci. Nous montrons ensuite comment, à l'aide de relations graphiques simples, nous identifions à partir de ces lignes les murs qui forment le document. Dans la section suivante, nous présentons le processus d'identification de la segmentation en pièces du plan d'habitation considéré à partir des murs qui ont été détectés. Nous nous appuyons sur un mécanisme de décomposition récursive de l'image jusqu'à l'obtention de régions convexes. Nous montrons comment la spécificité des documents architecturaux est prise en considération afin d'obtenir un ensemble de régions majoritairement rectangulaires correspondant aux pièces. Nous présentons ensuite des résultats quantitatifs obtenus sur un corpus de plans réels qui illustrent la pertinence de l'approche.

## 2. Détection de lignes et de murs

La première étape de notre méthode d'interprétation de plans d'habitation consiste en l'extraction des primitives graphiques intéressantes : les lignes. Notre algorithme de détection de lignes s'appuie sur le couplage de deux techniques bien connues : la transformée de Hough (Song *et al.*, 2002) et la vectorisation (Rosin *et al.*, 1989). L'algorithme de vectorisation permet d'approximer le contenu de l'image. Toutefois, la paramétrisation de ce processus est délicate et tend à fragmenter les éléments. Nous obtenons ensuite un ensemble plus compact de lignes en employant la transformée de Hough. Dans cette section, nous rappelons tout d'abord le principe de la transformée de Hough « classique ». Nous nous focalisons ensuite sur les bénéfices apportés par l'exploitation de cette technique en aval d'une vectorisation.

### 2.1. Transformée de Hough classique

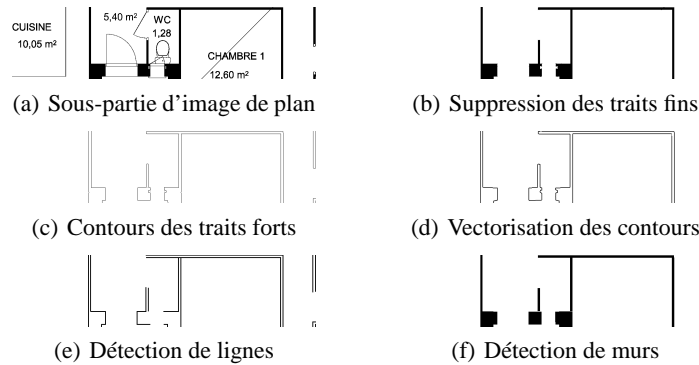
La transformée de Hough est une méthode bien connue pour la détection de caractéristiques dans les images ; elle a été énormément utilisée pour la détection de lignes. Le principe est de définir un espace des paramètres, ou *accumulateur*. Une ligne pouvant être définie par deux paramètres, sa distance  $\rho$  par rapport à l'origine et l'angle  $\theta$  du vecteur de l'origine au point de la ligne qui en est le plus proche, cet accumulateur est un tableau à deux dimensions dont chaque case représente une ligne potentielle du document. Chaque pixel noir de l'image (ou point caractéristique, point médian, etc.) « vote » pour toutes les lignes qui passent par sa position. Ainsi des points alignés vont voter pour une même ligne ; ce processus est appelé *accumulation*. L'étape suivante est une *détection de pics* dans l'accumulateur. Si une case  $(\rho, \theta)$  est sélectionnée, alors la ligne associée existe vraisemblablement dans le document.

Cette exploitation classique de la transformée de Hough a plusieurs inconvénients. Le premier est sa complexité, qui résulte de sa nature exhaustive. Le second est la nécessité de mettre en place un post-traitement de *vérification de ligne* afin de valider ou non une hypothèse et de détecter les coordonnées des extrémités de la ligne correspondante (seules les directions, i.e. les valeurs  $\rho$  et  $\theta$ , sont calculées par la transformée de Hough classique). Le troisième inconvénient majeur de cette méthode est sa sensibilité au bruit : un point va voter pour *toutes* les lignes qui peuvent passer par sa position, augmentant ainsi le vote de lignes qui sont en fait incohérentes.

## 2.2. Couplage de la transformée de Hough et de la vectorisation

Le principe de l'approche que nous avons choisie pour la détection de lignes est de limiter l'impact des trois inconvénients cités dans la section précédente en réalisant la transformée de Hough sur le résultat de la vectorisation de l'image considérée. L'idée générale est que les *segments* résultants de la vectorisation contiennent des informations très significatives au sujet des lignes du document. Par exemple, il est improbable qu'un segment horizontal fasse en fait partie d'une ligne verticale dans l'image. Par conséquent les points qui le constituent ne devraient pas voter pour les lignes verticales. Pour ceci, nous conservons le processus général d'accumulation : nous considérons les points constituant chaque segment résultant de la vectorisation (l'algorithme de Bresenham est exploité pour calculer chacun de ces points), et leur vote est exploité pour le remplissage de l'accumulateur. Cependant, un point donné ne votera plus pour toutes les lignes qui sont en adéquation avec sa position, mais seulement pour celles qui sont également en adéquation avec l'angle du segment auquel il appartient. Pour un segment d'angle  $a$ , nous avons choisi de limiter le vote des points qui le constituent aux angles compris dans  $[a - \pi/8, a + \pi/8]$  (la vectorisation réalisant une approximation plus ou moins fidèle de l'image, limiter les calculs à l'angle  $a$  n'est pas suffisant). Cette technique est le premier avantage de notre méthode de détection de lignes. D'une part, la complexité de ce processus est diminuée (dans notre cas, nous divisons par quatre le nombre d'angles considérés pour un point donné). D'autre part, le bruit induit par un point est réduit, puisqu'il ne votera plus pour des lignes incohérentes vis-à-vis du segment auquel il appartient.

Le second avantage majeur d'exploiter la transformée de Hough sur le résultat de la vectorisation de l'image est de permettre de valider une hypothèse de ligne simplement en détectant les segments qui sont associés à une direction  $(\rho, \theta)$  détectée. Pour ceci, nous définissons une zone autour de cette direction et cherchons les segments qui sont y localisés (seuls les segments avec un angle valide sont conservés). Le but est alors de reconstituer la séquence de segments formant la ligne. Afin de limiter la complexité de ce processus, une structure de données est exploitée afin d'accéder rapidement aux segments localisés à des coordonnées données. Deux segments valides consécutifs peuvent être ou non connectés. Il est alors nécessaire d'évaluer la distance entre ces segments pour déterminer s'ils définissent une même ligne ou si le premier termine une ligne et le second en commence une autre (un seuil est défini



**Figure 2.** Illustration des différentes étapes de notre méthode de détection de murs.

dans ce but). Enfin, les coordonnées des lignes détectées sont calculées par projection orthogonale des extrémités des segments qui la constituent sur la direction considérée. Un post-processus peut être mis en place de manière à éliminer les lignes qui ne sont pas jugées suffisamment longues (un autre seuil est nécessaire pour cela).

Le troisième avantage de cette nouvelle méthode est de limiter l'impact des votes des lignes qui ont été détectées. En effet, nous savons de quelle manière les points d'un segment ont voté et il est donc simple d'« annuler » ces votes. Par extension, enlever une ligne de l'image, et donc de l'accumulateur, consiste simplement à annuler les votes des segments qui la constituent. Nous appliquons ce principe à chaque segment constituant les lignes qui ont été détectées, ce qui supprime le bruit induit par cette ligne sur le reste du document. Ainsi, à chaque étape du processus, la détection de la ligne « suivante » consiste à trouver le « nouveau » maximum local dans l'accumulateur. Cette technique, couplée avec la limitation des angles considérés pour un point donné, limite de manière intéressante l'impact d'un point donné sur la détection des lignes de l'image.

### 2.3. Application à la détection des murs dans des plans d'architecture

Lors de l'analyse de plans architecturaux, nous mettons en place un ensemble de pré-traitements afin d'isoler les murs. Ces différents processus sont illustrés sur la figure 2 sur une sous-partie d'image de plan (figure 2(a)). Dans le cas des images que nous considérons, les murs sont représentés avec des lignes généralement plus épaisses que celles utilisées pour représenter des informations de natures différentes (dimensions, projections de toiture, etc.). Par conséquent, nous appliquons une séparation traits forts/trait fins suite à une séparation texte/graphique (figure 2(b)). Nous réalisons ensuite une détection de contours afin de ne pas être dépendant des performances de la vectorisation lors de l'analyse de traits forts, ces derniers étant générale-

ment sources de problèmes (figure 2(c)). Ces trois pré-traitements sont effectués à l'aide de techniques classiques (Rendek *et al.*, 2004).

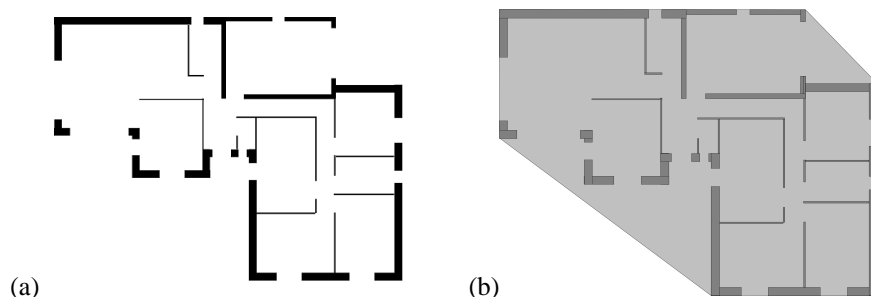
Une fois ces pré-traitements terminés, l'algorithme de détection de lignes est appliqué (le résultat est présenté figure 2(e)). Le but du processus de détection de murs est alors d'identifier des candidats de murs à partir des lignes détectées. Comme introduit précédemment, les lignes détectées sont supposées être les contours de ces murs. Ainsi, la première étape de ce processus consiste à identifier les lignes parallèles et proches. La seconde étape consiste à valider (pixels noirs ici) ou non ces hypothèses en analysant la texture localisée entre deux lignes. La séparation traits forts/trait fins s'apparente sur notre premier corpus à une détection de texture mais il conviendra par la suite d'identifier la texture employée. Un exemple de résultat de notre algorithme de détection de murs est présenté sur la figure 3(a).

### 3. Détection des pièces

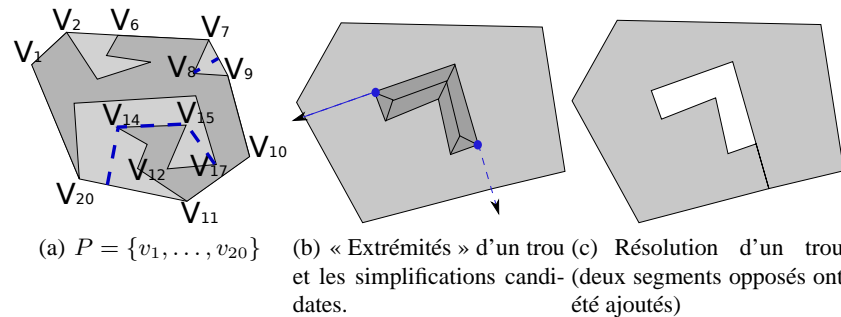
#### 3.1. Définition du problème

La détection d'une pièce d'un plan d'architecture peut être obtenue en identifiant exhaustivement l'ensemble des murs définissant son pourtour. À cette étape de notre projet, nous n'effectuons pas d'association entre cette première représentation et l'éventuel toponyme présent dans la zone ainsi définie. Ainsi, tel que le problème a été posé, la tâche de détection des pièces d'une habitation consiste à retrouver la meilleure partition de l'image analysée, pour laquelle chaque région correspond à une pièce. En observant les plans architecturaux, nous pouvons admettre que les pièces sont essentiellement des régions convexes. Nous retenons ainsi cette propriété pour définir un critère à optimiser durant le processus de segmentation (cf §3.2.1).

En considérant les murs détectés comme étant des rectangles, nous adoptons une représentation polygonale de la scène à analyser (l'enveloppe convexe des murs détectés, cf figure 3). Une approche descendante, exploitant un critère géométrique, identifie progressivement les séparations entre les pièces.



**Figure 3.** Pour le plan de la figure 1(b) : (a) murs détectés et (b) scène à décomposer



**Figure 4.** Notions mises en œuvre au sein de la décomposition approximativement convexe. L'enveloppe convexe de  $P$  est  $CH(P) = \{v_1, v_2, v_6, v_7, v_9, v_{10}, v_{11}, v_{20}\}$ , l'une des poches de  $P$  (parmi 3) est,  $\{v_{11}, \dots, v_{20}\}$  dont le pont est  $v_{11}v_{20}$ .

### 3.2. Décomposition polygonale

Dans le but de détecter les pièces constituant une habitation, nous faisons l'hypothèse qu'une pièce correspond à une région *approximativement convexe*. La notion de convexité, fréquemment binaire, doit être flexible et quantifiée. En adoptant la démarche décrite dans (Lien *et al.*, 2006), une telle décomposition peut être obtenue en supprimant récursivement les caractéristiques visuelles singulières, détectées au niveau de points de contour. Pour y parvenir, des éléments de séparation, des segments issus des points « singuliers », sont ajoutés.

Afin de présenter le principe général de l'algorithme de partitionnement de polygone<sup>1</sup>, considérons un polygone  $P = \{v_i\}_{i \in [1, N]}$  à  $N$  sommets, ne possédant aucun trou. Par la suite, nous nous appuierons sur la figure 4(a). Pour chaque sommet  $v_i$  de ce polygone, une mesure de concavité est associée. Si ce sommet est situé sur l'enveloppe convexe  $CH(P)$  de  $P$ , cette mesure est nulle. Quantifier la notion de concavité d'un point n'apparaît que pour les parties concaves de  $P$  appelées *poches*. Une telle poche est définie par l'ensemble des sommets consécutifs qui n'apparaissent pas dans la définition de  $CH(P)$ . Un *pont* est un segment de  $CH(P)$  liant deux sommets  $v_i$  et  $v_k$  non adjacents de  $P$ . La concavité d'un point  $v_i$  est alors caractérisée par une distance liant  $v_i$  au pont de la poche à laquelle  $v_i$  appartient. On qualifie dans ce contexte un polygone  $P$  de  $\tau$ -approximativement convexe si toutes les mesures de concavité de ses sommets sont inférieures à  $\tau$ .

#### 3.2.1. Caractérisation de la convexité / de la concavité

Une mesure de convexité nous permet d'évaluer si une région correspond potentiellement à une unique pièce, ou si elle doit être scindée. Les mesures de similarité entre une forme et son enveloppe convexe sur la base d'un critère de périmètre ou

1. Cf (Lien *et al.*, 2006) pour une présentation formelle de cette approche.



de surface sont peu pertinentes. En effet, ces mesures caractérisent globalement une forme à l'inverse d'une mesure de la profondeur des poches. La concavité d'un polygone est ainsi définie comme étant la valeur maximale de la distance séparant un point du pont de la poche à laquelle ce point appartient. Le plus court chemin au sein de la poche représente un choix intuitif et performant. Pour le polygone  $P$  de la figure 4(a), sa concavité est maximale au niveau du sommet  $v_{17}$  dont le plus court chemin jusqu'au pont  $v_{11}v_{20}$  est la séquence  $v_{17}, v_{15}, v_{14}, \text{proj}_{\perp}(v_{14}, [v_{11}v_{20}])$ . En présence d'un sommet  $v^*$  associé à une mesure de concavité supérieure à un seuil, un polygone doit être scindé en sélectionnant une diagonale issue de  $v^*$ .

### 3.2.2. Sélection d'une coupure

Lorsqu'un polygone  $P$  n'est pas considéré convexe, celui-ci est scindé en employant un segment  $v_b v_e$  issu du sommet  $v_b \in P$  qui maximise la mesure de concavité. L'autre extrémité  $v_e \in P$  de cette diagonale est déterminée en optimisant une fonction appropriée au domaine applicatif. Dans notre cas, puisque les pièces sont principalement des formes rectangulaires,  $v_e$  est déterminé en minimisant l'expression suivante :

$$\sin(\max \{ \theta_{b-1,b,k}, \theta_{k,b,b+1}, \theta_{b,k,k+1}, \theta_{b,k,k-1} \}) \| \overrightarrow{v_k v_b} \|$$

où  $[v_k v_b] \cap P = \emptyset$ , les sommets  $v_k$  et  $v_b$  se « voient », ils forment une diagonale de  $P$ , et

$$\theta_{a,b,c} = \min \left\{ \pi \pm \widehat{v_a, v_b, v_c}, \frac{\pi}{2} \pm \widehat{v_a, v_b, v_c} \right\}$$

Une telle expression permet de favoriser les diagonales courtes ainsi que les angles plats et angles droits autour des extrémités de cette séparation entre les régions.

### 3.2.3. Partitionnement en présence de trous

En présence de murs non connectés au restant de l'habitation, le polygone  $P$  possède des trous (cf figure 3(b), le problème est initialisé en considérant l'enveloppe convexe des murs). Leurs existences impliquent en premier lieu de les relier à la bordure externe de  $P$ . Lorsqu'un polygone  $P$  possède un trou  $H$ , une simplification intuitive consiste à relier ce trou à la bordure externe de  $P$  en sélectionnant la direction principale observée autour de l'une des deux « extrémités » de  $H$  (cf figure 4(b)). Ces points correspondent à la paire de points associés à la géodésique de  $H$ . Nous obtenons une estimation de cette paire de points à partir du squelette rectiligne de  $H$  (cf figure 4(b)) et retenons le liens le plus court (cf figure 4(c)).

Concernant le coût algorithmique de cette décomposition, si nous omettons le calcul de la mesure de concavité, la décomposition d'un polygone  $P$  à  $N$  sommets et ayant  $k$  trous en  $m$  régions nécessite  $\mathcal{O}(Nr)$  opérations, avec  $r = m + k$ .

Comme le précise Lien et Amato dans leurs travaux, la pertinence de la décomposition repose notamment sur l'ordonnancement des sommets des poches. En outre, dans la mesure où nous travaillons sur les données issues de la vectorisation des murs, certaines séparations intuitives peuvent ne pas avoir été sélectionnées. En effet, dès que le

premier point d'une séparation a été identifié grâce à sa concavité, seuls les sommets initialement présents dans la définition de  $P$  (enveloppe externe et trous compris) sont considérés comme des extrémités candidates. À titre d'exemple, seules les diagonales  $[v_{10}v_{17}]$  et  $[v_{11}v_{17}]$  sont évaluées pour décomposer le polygone  $P$  de la figure 4(a) où  $v_{17}$  maximise la concavité. Au lieu d'introduire durant ce processus des points supplémentaires, nous avons défini certaines règles en fonction de notre domaine applicatif dans le but d'obtenir des formes rectangulaires pour chaque pièce.

L'approche est implémentée en s'appuyant sur l'algorithme de (Hipke, 1996) pour le calcul des relations de visibilité, en l'adaptant afin de prendre en considération les configurations dégénérées (e.g. colinéarité de sommets, polygones non simples) avec la bibliothèque (CGAL, n.d.).

### 3.3. Filtrage des régions détectées

Pour employer une décomposition polygonale du logement, nous admettons qu'une bonne estimation du contour de l'habitation est l'enveloppe convexe des murs précédemment détectés. L'étape précédente garantit que chaque région proposée est  $\tau$ -approximativement convexe. Toutefois, lorsque le logement ne possède pas une forme strictement rectangulaire, le problème est initialement mal formulé (cf figure 3(b)). Pour y remédier, nous filtrons ce résultat de segmentation en retirant itérativement les régions périphériques qui ne semblent pas rectangulaires ou dont les ouvertures (les contours qui ne correspondent pas à un mur) sont trop grandes. Pour cette étude, nous identifions le rectangle de référence d'un polygone afin d'établir s'il s'agit ou non d'une pièce.

Concernant l'identification de cette forme de référence, nous avons adapté le principe du *Rectangle Enveloppant d'Aire Minimale* (Freeman *et al.*, 1975) en optimisant la fonction ci-dessous pour  $\mathcal{V} = \{sv_i\} \subseteq \mathcal{S}$ , l'ensemble des segments dont les deux extrémités voient les contours de l'enveloppe convexe de la forme. L'orientation de ce rectangle correspond à celle du segment  $sv^*$  :

$$sv^* = \arg \max_{sv_i} \sum_{sv_j \in \mathcal{V}} \min\{|\vec{w}_i \cdot \vec{sv}_j|, |\vec{h}_i \cdot \vec{sv}_j|\}$$

où  $\vec{w}_i$ , respectivement  $\vec{h}_i$ , désigne le vecteur colinéaire, respectivement orthogonal, à  $\vec{sv}_i$ . En employant cette formulation, nous déterminons le rectangle dont la somme totale des longueurs des segments visibles projetés est maximale. Restreindre  $\mathcal{S}$  à  $\mathcal{V}$  a deux intérêts : i. les cloisons internes d'une pièce sont considérés comme des détails et ii. seuls les contours externes correspondant précédemment à la détection des murs sont fiables, dans la mesure où leurs détections est plus fiable que le processus de sélection des diagonales lors de la décomposition polygonale (cf remarque sur la détermination d'une coupure au § précédent).

	Pièces présentes	Pièces détectées	1 - *	* - 1	Taux de détection	Taux de reconnaissance
Moyenne	8.93	17.55	2.31	0.47	0.85	0.63
Écart type	2.26	5.71	1.21	0.93	0.12	0.12

**Tableau 1.** Performances obtenues en suivant le protocole de Phillips et Chhabra. Les colonnes « 1 - \* » et « \* - 1 » concernent les mises en correspondances multiples entre la vérité terrain d'une part et les résultats obtenus d'autre part.

## 4. Évaluations

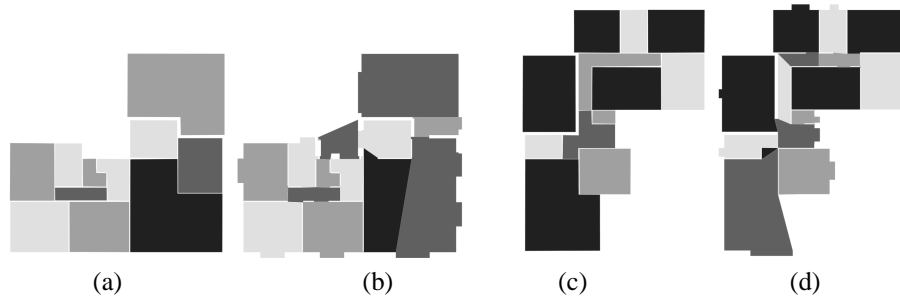
### 4.1. Corpus de plans d'habitation

Afin d'évaluer de manière réaliste la méthode que nous proposons, nous nous sommes procurés un ensemble de documents auprès d'un cabinet d'architectes. Ces documents couvrent, pour ce cabinet, une période de plus de dix ans, et sont donc supposés représenter les variations qui ont pu avoir lieu dans les conventions et les constructions. Il s'agit à la fois de plans de maisons et de plans d'appartements.

Les images originales sont des images en couleurs, celles-ci étant généralement utilisées pour différencier les pièces les unes des autres ; elles explicitent donc la localisation des pièces. Nous n'avons pas exploité ces informations pour notre prototype afin de rester aussi générique que possible. En effet, l'emploi de couleurs est relativement rare pour les plans papiers. Nous avons travaillé sur le résultat de la binarisation de ces images. Cependant, nous avons exploité ces couleurs pour notre étiquetage, puisqu'elle définissent notre vérité terrain. Nous avons ainsi pu prendre la pleine mesure de la difficulté et de la subjectivité de la tâche : il est parfois impossible, en considérant uniquement l'image binarisée, de déduire les limites entre une pièce et une autre (essentiellement en présence de « cuisines ouvertes », voir figure 1).

### 4.2. Protocole expérimental et résultats

La tâche d'interprétation semi-automatique de documents architecturaux doit conduire dans cette étude à une saisie préalable des principales entités manipulées par les utilisateurs, les pièces. Idéalement, l'ensemble des pièces doit être retranscrit automatiquement. Tant le nombre de pièces que la description individuelle de chaque pièce (emplacement des murs, des ouvertures) sont sujets à quelques erreurs que l'utilisateur devra ensuite corriger manuellement. Afin de quantifier les erreurs produites par notre système, nous avons adopté le protocole établi dans (Phillips *et al.*, 1999). Ce dernier a le mérite de faire apparaître les erreurs de segmentation individuellement. En ce sens, les critères de surface ne sont employés qu'au moment de la recherche du meilleur appariement entre les régions identifiées lors de la saisie de la vérité terrain et celles proposées par notre prototype. À ce stade de notre projet, nous estimons que seul le nombre d'interactions avec l'utilisateur doit être quantifié, laissant de côté



**Figure 5.** Pièces présentes (a,c) et détectées (b,d) pour les plans de la figure 1. Les couleurs employées n'ont pour seul but que de simplifier la lecture des résultats.

pour le moment le coût, subjectif ou non, de chacune de ces éditions. À partir de 80 plans sommaires, pour lesquels les informations de cotation sont notamment absentes, nous obtenons les performances du tableau 1 en ce qui concerne l'identification des pièces. Comme ce tableau l'indique, nous sur-segmentons la plupart des documents (colonne 1-★). Bien qu'il convienne de réduire ce phénomène, il nous semble plus aisé pour un utilisateur d'effectuer une fusion que d'indiquer précisément l'emplacement d'une séparation. L'étape de filtrage des pièces, dans le but de supprimer les régions externes au logement, nous amène à supprimer certaines pièces ou parties de pièces (voir figure 5). La séparation des pièces ouvertes est une tâche difficile en l'état. La couche textuelle, sur laquelle apparaît en particulier les côtes et la superficie, pourrait orienter certains choix.

## 5. Conclusion

Dans cet article, nous avons présenté une nouvelle méthode d'interprétation de plans d'habitation, et plus spécifiquement d'extraction de leurs pièces. Notre approche est basée sur deux étapes principales, qui sont l'extraction des primitives du document (les lignes formant les murs du logement) et la détection des différentes pièces qui le constituent. En ce qui concerne la détection de lignes, nous nous sommes appuyés sur une technique originale consistant à coupler une méthode classique basée sur la transformée de Hough avec une vectorisation de l'image. Le but est que chacun de ces deux processus puissent s'entraider de manière à produire une détection de lignes à la fois robuste et efficace. Des propriétés graphiques sont ensuite exploitées de manière à déduire, à partir de ces lignes, la localisation des murs du document. En ce qui concerne la détection des pièces, nous avons exploité une décomposition descendante, récursive, de l'image, consistant à identifier des régions peu homogènes, et ce jusqu'à l'obtention de régions quasi convexes.

Ces premiers résultats ont fait l'objet d'une évaluation sur un corpus de données réelles, obtenues auprès d'architectes. Nous avons effectué un étiquetage manuel de

Hervé Locteau, Sébastien Macé, Salvatore Tabbone, Ernest Valveny

la localisation des pièces, et avons ainsi mis en place des expérimentations dont les résultats montrent la pertinence de la méthode proposée.

Il existe de nombreuses perspectives à ces travaux. La première est bien sûr l'évaluation de sa généralité pour déterminer dans quelles limites cette approche peut être exploitée. Il s'agira alors de proposer des extensions de manière à accroître la palette de conventions graphiques interprétables. La seconde perspective est de prendre plus fortement en compte l'utilisateur final du système. Cela va tout d'abord nécessiter la conception d'une interface graphique permettant à l'utilisateur de corriger les erreurs du système, de manière à obtenir finalement un document parfaitement interprété. Nous souhaiterions alors mettre en place des évaluations plus fines afin d'évaluer les impacts des différentes prises de décision au sein de notre approche (par exemple en termes de sur/sous-segmentation en pièces) sur la nature des corrections nécessaires. Finalement, nous réfléchirons à la mise en place d'une évaluation qui soit plus directement en adéquation avec l'effort qui sera demandé à l'utilisateur en post-traitement.

### Remerciements

Ces travaux sont financés par l'Union Européenne (projet EUREKA 4462 ScanPlan) et le ministère de l'industrie espagnol (projet AVANZA I+D TSI-020400-2009-115).

## 6. Bibliographie

- CGAL, « Computational Geometry Algorithms Library », n.d. <http://www.cgal.org>.
- Dosch P., Tombre K., Ah-Soon C., Masini G., « A complete system for the analysis of architectural drawings », *IJDAR*, vol. 3, n° 2, p. 102-116, 2000.
- Freeman H., Shapira R., « Determining the minimum-area encasing rectangle for arbitrary closed curve », *Communications of ACM*, vol. 18, p. 409-413, 1975.
- Hipke C. A., « Computing Visibility Polygons with LEDA », 1996.
- Lien J.-M., Amato N. M., « Approximate convex decomposition of polygons », *Computational Geometry*, vol. 35, p. 100-123, 2006.
- Lladós J., López-Krahe J., Martí E., « A system to understand hand-drawn floor plans using subgraph isomorphism and Hough transform », *Machine Vision and Applications*, vol. 10, n° 3, p. 150-158, 1997.
- Phillips I., Chhabra A. K., « Empirical Performance Evaluation of Graphics Recognition Systems », *IEEE PAMI*, vol. 21, n° 9, p. 849-870, 1999.
- Rendek J., Masini G., Dosch P., Tombre K., « The search for genericity in graphics recognition applications : Design issues of the qgar software system », *IAPR International Workshop on Document Analysis Systems*, vol. 3163 of LNCS, Springer Verlag, p. 366-377, 2004.
- Rosin P. L., West G. A. W., « Segmentation of edges into lines and arcs », *Image Vision Comput.*, vol. 7, n° 2, p. 109-114, 1989.
- Song J., Cai M., Lyu M. R., Cai S., « A New Approach for Line Recognition in Large-size Images Using Hough Transform », *ICPR*, vol. 1, p. 10033, 2002.