

Génération automatique d'un modèle de bâtiment à partir d'un croquis

Jean-Noël Demaret^{*} et Pierre Leclercq

LUCID, Université de Liège

Chemin des Chevreuils 1 Bât. B52/3, 4000 Liège, Belgique

*Aspirant F.R.S.-FNRS

RÉSUMÉ. Les recherches en interprétation automatique de croquis ouvrent la voie à l'émergence de nouveaux outils capables de combiner la souplesse du croquis à main levée avec la puissance des outils numériques d'évaluation et de simulation des performances d'un bâtiment. Un croquis contient par nature un grand nombre d'ambiguités et son interprétation implique dès lors l'exploration d'un espace d'interprétation qui peut rapidement devenir gigantesque. Dans cet article, nous présentons NEMO, un outil d'interprétation de croquis d'architecture basé sur un système multi-agent adaptatif qui lui permet d'explorer plus efficacement cet espace des interprétations possibles.

MOTS-CLÉS : interprétation de croquis, système multi-agent, système adaptatif.

1. Introduction

Les architectes ont aujourd'hui accès à de nombreux outils informatiques d'évaluation et de simulation des performances d'un bâtiment pour les assister dans leur travail de conception. Ces outils sont cependant difficilement exploitables dès la phase préliminaire d'un projet. En effet, les logiciels classiquement utilisés pour créer le modèle numérique d'un bâtiment contraignent le concepteur à encoder un modèle complet et extrêmement défini d'un objet architectural initialement flou ; ils s'intègrent dès lors difficilement dans le flux d'un processus créatif. En outre, le temps requis par cette étape d'encodage rend laborieuse leur exploitation dans un travail itératif de conception.

A contrario, le croquis à main levée offre au concepteur une totale liberté d'expression et une grande souplesse d'utilisation. Il s'avère ainsi être un outil incontournable pour soutenir l'idéation et reste universellement utilisé dans tous les domaines de la conception, en particulier dans celui de l'architecture. L'esquisse se révèle ainsi être le support le plus adapté d'externalisation et de communication de la sémantique d'un projet architectural (Leclercq, 1997).

Les recherches en interprétation automatique de croquis visent à concevoir des systèmes capables d'analyser un croquis et de générer une représentation signifiante (un modèle) traduisant l'intention du dessinateur. Dans le cadre de l'assistance à la conception, l'objectif poursuivi par ces travaux est de permettre l'utilisation d'outils numériques d'évaluation et de simulation dès la phase de conception précoce, quand le projet n'existe encore que sous la forme d'une ou plusieurs esquisses.

Un certain nombre de systèmes de reconnaissance de croquis ont déjà été proposés : citons, par exemple, (Gross & Do, 1997) pour l'utilisation du croquis comme moyen d'interrogation d'une base de connaissances ; (Coyette *et al.*, 2004) pour la génération d'une interface utilisateur à partir d'une esquisse ; et (Casella *et al.*, 2008a) pour la reconnaissance de diagrammes UML à main levée.

Dans le domaine de l'architecture, quelques prototypes logiciels ont déjà pu démontrer la faisabilité et le potentiel des systèmes d'aide à la conception utilisant le croquis. Citons, parmi eux, EsQUIsE (Leclercq, 2004), un outil capable de reconstruire le modèle tridimensionnel d'un bâtiment esquissé et de produire une estimation de ses besoins calorifiques. Le prototype ICC (Safin *et al.*, 2007) permet quant à lui la mise au net et la détection d'incohérences dans des relevés à main levée réalisés sur site. Enfin, Casella *et al.* (2008b) proposent un environnement d'analyse de croquis, avec comme application la vérification du respect de normes de sécurité dans l'aménagement intérieur d'un immeuble.

Malgré les perspectives qu'ils ouvrent, aucun de ces systèmes n'a encore été adopté dans le monde professionnel. D'une part, la mise en œuvre d'interfaces logicielles basées sur des interactions au stylo et utilisant le croquis pose encore de nombreuses questions d'ergonomie (Safin, 2011). D'autre part, le problème de l'interprétation automatique d'esquisses à main levée est loin d'être résolu d'un point de vue technique : les prototypes existants souffrent encore de nombreux problèmes de robustesse qui rendent impossible leur utilisation dans des conditions réelles. C'est à ce deuxième axe de recherche que nous nous intéresserons dans cet article.

2. Problématique de l'interprétation de croquis

A l'instar de la compréhension du langage naturel, l'interprétation d'un croquis est une tâche qui s'avère extrêmement complexe à implémenter sur un ordinateur. Cette complexité trouve sa source dans la nature intrinsèquement ambiguë d'un croquis et dans la multiplicité des connaissances qui doivent être mises en œuvre pour le comprendre.

Un important facteur d'ambiguités est l'imprécision inhérente au dessin à main levée. Il peut ainsi être délicat de déterminer l'existence d'une connexion entre deux traits. De même, les variations imprévisibles causées par ces imprécisions rendent complexe la mise au point de modules de reconnaissance de symboles fiables.

Un second facteur d'ambiguités est la présence de significations implicites. Un cercle dans une esquisse d'architecture pourra ainsi représenter une lettre « O », une chaise... ou un carré grossièrement dessiné.

La plupart du temps, la résolution d'une ambiguïté nécessitera la prise en compte de son contexte spatial ; mais comme ce dernier peut lui-même contenir des ambiguïtés, l'espace d'interprétation d'un croquis (l'ensemble des interprétations possibles) peut rapidement devenir gigantesque.

L'interprétation d'un croquis implique, d'une part, une tâche de segmentation au cours de laquelle des traits, ou des segments de traits, sont regroupés pour former des objets distincts, et, d'autre part, une tâche de reconnaissance qui consiste à identifier chacun de ces objets comme une entité signifiante (un symbole) et à lui attribuer l'étiquette correspondante. La tâche globale d'interprétation peut ainsi être formalisée comme un problème de recherche dans lequel l'objectif est de trouver, dans l'espace de tous les groupements possibles et de tous les étiquetages possibles de ces groupements, l'ensemble des groupements et des étiquetages correspondants à l'intention du dessinateur (Shilman *et al.*, 2004).

En utilisant ce cadre des problèmes de recherche, il est possible de comparer les systèmes d'interprétation existants en étudiant la façon dont ils explorent l'espace d'interprétation d'un croquis. Ainsi, dans certains de ces systèmes (Ramel *et al.*, 1998 ; Juchmes *et al.*, 2005), les ambiguïtés rencontrées sont résolues immédiatement et seule la meilleure hypothèse, en fonction du contexte courant, est conservée. Ces systèmes ne peuvent dès lors pas tenir compte des modifications ultérieures de ce contexte : chaque décision est définitive et n'est jamais remise en cause. D'autres systèmes (Hammond & Davis, 2005 ; Plimmer & Freeman, 2007 ; Casella *et al.*, 2008a) procèdent à une exploration exhaustive de l'espace d'interprétation. Les ambiguïtés sont alors résolues à la fin du processus par

la sélection d'un sous-ensemble d'hypothèses cohérentes pour construire l'interprétation finale du croquis. A cause de cette approche exhaustive, ces systèmes doivent limiter le nombre d'interprétations envisagées en utilisant des hypothèses d'analyse qui, au final, restreignent la liberté du dessinateur. Le système proposé dans (Macé & Anquetil, 2009) force, par exemple, l'utilisateur à terminer le dessin d'un symbole avant de commencer le suivant. Le système SketchREAD (Alvarado & Davis, 2004) propose quant à lui une approche plus sophistiquée dans laquelle seules les hypothèses les plus prometteuses sont initialement générées. Le système est ensuite capable d'utiliser le contexte pour corriger certaines erreurs de reconnaissance de bas niveau. L'approche utilisée ne s'applique cependant qu'aux croquis de type « diagramme » dont les symboles peuvent être entièrement décrits à partir de primitives géométriques. Le système ne peut, par exemple, pas reconnaître des annotations textuelles.

3. NEMo

NEMo est un prototype expérimental dédié à la saisie et à l'interprétation de croquis à main levée représentant des plans d'étages en deux dimensions. Le cœur du logiciel réside dans son moteur d'interprétation qui est capable d'analyser un croquis précédemment capturé, de reconnaître les entités graphiques qu'il contient et de générer un aperçu 3D du bâtiment qui peut ensuite être exporté sous la forme d'un modèle OBJ (voir figure 1). Les entités actuellement reconnues sont les murs, les portes, les fenêtres, un ensemble de symboles de mobilier et certaines annotations.

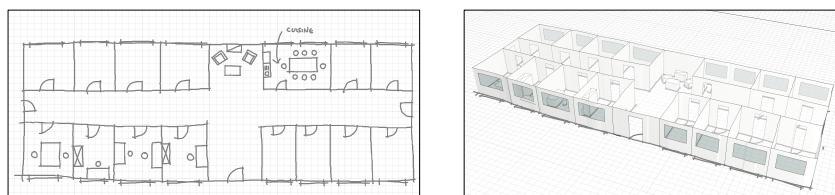


Figure 1 : Deux captures d'écran issues du prototype NEMo : un croquis capturé (à gauche) et l'aperçu 3D généré suite à son interprétation (à droite).

Dans la suite de cette section, nous présenterons les principales caractéristiques de NEMo, dont son architecture logicielle, et expliquerons en quoi elles le diffèrent des systèmes existants, en particulier de son précurseur EsQUIsE (Leclercq, 2004).

3.1. Interprétation asynchrone

NEMO se distingue d'abord de la plupart des systèmes d'interprétation récemment proposés par son fonctionnement asynchrone. Il a en effet été conçu pour interpréter un croquis terminé et non pour effectuer l'interprétation continue et incrémentale d'un croquis en cours de réalisation. Ce choix de conception se justifie par plusieurs observations.

Premièrement, la reconnaissance *online* est principalement utile dans les logiciels d'encodage orienté stylo, quand un retour immédiat est nécessaire afin de permettre à l'utilisateur une identification et une correction instantanée des erreurs de reconnaissances (Macé & Anquetil, 2009). Dans un scénario d'aide à la conception préliminaire, il est au contraire préférable que le processus de reconnaissance n'interfère pas dans le processus créatif (Safin, 2011).

Deuxièmement, l'interprétation *online* est essentiellement exploitable et intéressante du point de vue des performances pour l'analyse de croquis composés de symboles distincts qui peuvent être reconnus les uns après les autres au fur et à mesure qu'ils sont dessinés. Or, cette caractéristique n'est pas présente dans les croquis d'architecture qui sont composés d'entités graphiques fortement interconnectées.

Enfin, la nature incrémentale de l'interprétation *online* complique la gestion des opérations d'édition du croquis, comme l'effacement d'un trait ou la modification d'un symbole, et la plupart des systèmes *online* ne les supportent pas, ce qui les rend peu compatibles avec un processus d'idéation.

3.2. Segmentation perceptuelle

La plupart des systèmes d'interprétation de croquis existants visent la reconnaissance de diagrammes à main levée tels que des diagrammes UML (Casella *et al.*, 2008a) ou des schémas représentant des circuits électriques (Alvarado & Davis, 2004 ; Macé & Anquetil, 2009). Ces croquis ont la particularité d'être constitués de symboles isolés reliés entre eux par des connecteurs. Afin de limiter le nombre d'hypothèses de segmentation à envisager, ces systèmes font l'hypothèse simplificatrice que ces symboles et ces connecteurs sont exclusivement composés d'ensembles de traits disjoints. Les hypothèses de segmentation sont dès lors formées par regroupement de traits temporellement ou spatialement proches.

Bien qu'elle contraine la liberté du dessinateur, cette hypothèse est acceptable et se vérifie la plupart du temps dans le cas des diagrammes

(Alvarado & Davis, 2004). Elle est par contre incompatible avec l'interprétation de croquis d'architecture dans lesquels un trait ou un segment de trait sera fréquemment partagé par plusieurs symboles : un symbole représentant une table placée contre un mur ou contre un autre meuble partagera ainsi un (sous-)trait commun avec ce mur ou ce meuble.

La présence de traits partagés complexifie la tâche de segmentation car la nécessité de prise en considération des sous-trait, qui sont par définition bien plus nombreux que les traits, et la suppression de la contrainte de non partage, conduisent à une explosion combinatoire du nombre d'hypothèses de segmentation possibles. Ainsi, des systèmes comme (Leclercq 2004) ou (Casella *et al.*, 2008b), bien qu'ils visent l'interprétation de croquis architecturaux, ne sont pas capables de les gérer.

Pour répondre à ce problème, NEMo utilise une approche différente pour la segmentation, qui prend pour point de départ l'identification de l'importance de la notion de zone dans un croquis d'architecture. En effet, la subdivision et l'organisation de l'espace sont des éléments importants du travail de conception architecturale et le concept de zone, bien plus que la notion de traits, se révèle primordial dans la compréhension d'un croquis d'architecture. Ainsi, la méthode de segmentation principale implantée par NEMo ne repose pas sur le regroupement de traits mais sur l'identification de régions compatibles avec un ensemble d'heuristiques perceptuelles (Wuersch & Egenhofer, 2008).

Ce choix, plus en accord avec la nature d'un croquis architectural, permet non seulement au système d'effectuer une segmentation plus efficace du croquis grâce à l'identification directe des contours des espaces et des symboles, mais également de prendre en considération la présence de traits partagés.

3.3. Architecture logicielle

Le modèle informatique qui sous-tend NEMo s'inspire de l'architecture du programme Copycat (Mitchell, 2001), une architecture cognitive dont le domaine d'application initial est la découverte d'analogies entre des chaînes de lettres. Ce modèle, que nous allons décrire dans la suite de cette section, permet une exploration plus flexible de l'espace des hypothèses en n'écartant a priori aucune possibilité mais en affectant plus de ressources à l'exploration des hypothèses les plus probables. Il permet également, grâce à son comportement adaptatif, l'intégration des influences contextuelles dans le processus d'interprétation.

3.3.1. Architecture multi-agent

L'architecture de NEMo utilise le paradigme multi-agent. Les connaissances nécessaires à la construction des hypothèses d'interprétation sont ainsi distribuées entre différents modules logiciels, appelés *agents*, qui vont coopérer et rivaliser pour construire progressivement l'interprétation globale du croquis. Différents agents seront, par exemple, responsables de la segmentation des traits, de la reconnaissance des symboles ou des annotations textuelles. Cette modularité autorise l'utilisation de techniques de reconnaissance hétérogènes pour reconnaître différents types d'éléments comme des symboles ou du texte (Plimmer & Freeman, 2007 ; Casella *et al.*, 2008a). Le système est également capable d'utiliser en parallèle différentes stratégies pour accomplir la même tâche afin d'améliorer sa robustesse par rapport aux variations rencontrées dans un croquis. Ainsi, même si la segmentation s'effectue principalement par extraction des régions perceptuelles, d'autres méthodes comme l'identification des composantes connexes et le regroupement de traits, sont utilisées simultanément.

3.3.2. Espace de travail partagé

Les hypothèses d'interprétation sont construites par les agents dans un espace de travail global. Cette structure partagée permet une communication indirecte entre les agents : les hypothèses construites par un agent peuvent ainsi utiliser les hypothèses produites par d'autres agents, les renforcer, ou encore entrer en compétition avec elles. Par exemple, un agent de segmentation pourra supprimer (virtuellement) les traits appartenant à des symboles déjà reconnus par d'autres agents pour se créer une vue simplifiée du croquis et faciliter ainsi sa tâche de segmentation.

Cet espace de travail supporte de plus la définition de relations entre hypothèses. Une relation peut être une relation de compétition entre des hypothèses incompatibles, une relation d'agrégation ou une relation de renforcement contextuel, par exemple, entre une table et une chaise.

Les hypothèses incompatibles sont constamment en compétition par l'intermédiaire d'une procédure qui sélectionne régulièrement un sous-ensemble d'hypothèses cohérentes. Celles qui sortent vainqueurs de cette sélection gagnent de l'activation, tandis que les autres en perdent. Quand l'activation d'une hypothèse tombe à zéro, l'hypothèse est retirée de l'espace de travail. Ce mécanisme de sélection, qui s'inspire de la mét-heuristique d'optimisation par colonies de fourmis, a pour objectif

l'émergence progressive d'un sous-ensemble d'hypothèses formant une interprétation globale et cohérente du croquis.

3.3.3. Comportement adaptatif

Une caractéristique importante de NEMo est son comportement adaptatif. Contrairement aux systèmes déterministes classiques, ce comportement n'est pas prédéterminé mais dépend d'une population dynamique d'agents de traitement qui vont analyser le croquis et construire des hypothèses d'interprétation. Chaque agent est associé à une valeur de priorité qui détermine la vitesse à laquelle la tâche qu'il implémente va être exécutée. Ce mécanisme permet aux hypothèses plus prometteuses d'être explorées plus rapidement. Il autorise également l'intégration d'influences contextuelles. Par exemple, si une hypothèse de lettre — qui fait probablement partie d'un mot — est instanciée, des agents qui auront pour objectif de rechercher d'autres lettres proches de la première seront ajoutés dans le système ; ces agents augmenteront dès lors la probabilité que d'autres lettres soient découvertes dans le voisinage.

L'évolution de la population d'agents de traitement dépend d'un ensemble d'agents de connaissances qui vont réagir à l'instanciation de nouvelles hypothèses dans l'espace de travail en ajoutant des agents de traitement dans le système. Ceux-ci peuvent être des agents qui essaieront d'utiliser l'hypothèse préalablement instanciée pour construire des hypothèses de plus haut niveau, ou des agents qui chercheront des entités qui lui sont contextuellement liées. Ces derniers permettent d'initier localement une recherche plus approfondie pour localiser un symbole attendu, par exemple, en utilisant des seuils de reconnaissance plus faibles.

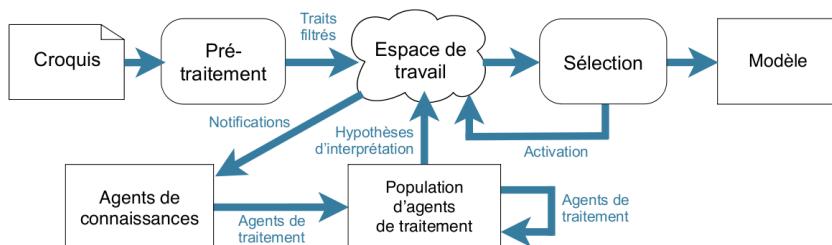


Figure 2 : Fonctionnement général de NEMo.

La figure 2 illustre le fonctionnement général de l'ensemble du modèle. Au démarrage du processus d'interprétation, la population d'agents de

traitement est initialisée avec un ensemble d'agents qui auront pour tâche de construire les premières hypothèses de segmentation. Quand une nouvelle hypothèse est instanciée, une notification est envoyée à chaque agent de connaissances, lequel, si l'hypothèse correspond à ses intérêts, réagit en ajoutant de nouveaux agents de traitement dans le système. Parallèlement, un processus de sélection est régulièrement exécuté afin de faire émerger progressivement une interprétation globale et unique du croquis. Finalement, un processus réflexif permet au système de surveiller sa propre activité et de mettre fin au processus d'interprétation dès que l'espace de travail atteint un état de stabilisation.

4. Conclusion

Nous avons présenté NEMo, un prototype expérimental capable d'interpréter un croquis à main levée représentant des plans d'étages et de générer un modèle 3D du bâtiment esquissé. Son implémentation est actuellement en cours de finalisation et des tests utilisateurs systématiques doivent encore être réalisés afin d'évaluer ses performances et sa robustesse. Cependant, grâce aux choix de conception qui ont guidé son développement et à son architecture logicielle interne, NEMo est déjà capable de surmonter certaines limitations de son prédecesseur EsQUIsE. Ainsi, son fonctionnement asynchrone lui permet d'abolir toute contrainte chronologique, comme l'obligation de terminer un symbole avant de commencer le suivant, et de supporter les opérations de modification du croquis qui s'avèrent indispensables au support du travail créatif. En utilisant une méthode de segmentation basée sur l'identification de régions perceptuelles, NEMo est également capable de reconnaître des symboles malgré la présence de traits partagés. Il ne constraint dès lors pas l'architecte à utiliser un mode inhabituel de représentation, notamment pour le dessin des équipements. Finalement, son architecture logicielle lui permet de mettre en œuvre une exploration plus flexible de l'espace d'interprétation d'un croquis. Il est ainsi capable, à la différence d'EsQUIsE, de gérer la coexistence d'hypothèses incompatibles et peut ainsi explorer simultanément différentes interprétations possibles. Il peut en outre exploiter les hypothèses déjà identifiées pour adapter son comportement afin d'examiner en priorité les solutions les plus prometteuses, avec comme objectif final d'explorer plus efficacement l'espace des interprétations possibles.

Bibliographie

- Alvarado, C., & Davis, R. (2004). SketchREAD: A Multi-Domain Sketch Recognition Engine. *Proc. of UIST 2004*, 23-32.
- Casella, G., Deufemia, V., Mascardi, V., Costagliola, G., & Martelli, M. (2008a). An agent-based framework for sketched symbol interpretation. *Journal of Visual Languages and Computing*, 19, 225-257.
- Casella, G., Deufemia, V., Mascardi, V., Martelli, M., & Tortora, G. (2008b). Resoning about Hand-Drawn Sketches: An Approach Based on Intelligent Software Agents. *LNCS 5188*, 302-314.
- Coyette, A., Faulkner, S., Kolp, M., Limbourg, Q., & Vanderdonkt, J. (2004). SKetchiXML: Towards a Multi-Agent Design Tool for Sketching User Interfaces Based on USIXML. *Proc. of TAMODIA 2004*.
- Gross, M. D., & Do, E. (1996). Demonstrating the Electronic Cocktail Napkin. *Proc. of the ACM Conference on Human Factors in Computing (CHI '96)*.
- Hammond, T., & Davis, R. (2005). LADDER, a sketching language for user interface developers. *Computers & Graphics*, 29, 518-532.
- Juchmes, R., Leclercq, P., & Sleiman, A. (2005). A freehand-sketch environment for architectural design supported by a multi-agent system. *Computers & Graphics*, 29(6), 905-915.
- Leclercq, P. (1997). From an architectural sketch to its semantic representation. *International Journal of Construction Information Technology*, 4(2), 67-84.
- Leclercq, P. (2004). Invisible Sketch Interface in Architectural Engineering. *Graphics Recognition, Recent Advanced and Perspectives, LNCS 3088*.
- Macé, S., & Anquetil, E. (2009). Eager interpretation of on-line hand-drawn structured documents: The DALI methodology. *Patter Recognition*, 42.
- Mitchell, M. (2001). Analogy-Making as a Complex Adaptive System. In L.A. Segel, & I. Cohen (Eds.), *Design principles for the immune system and other distributed autonomous systems*, 335-360.
- Plimmer, B., & Freeman, I. (2007). A Toolkit Approach to Sketched Diagram Recongition. *Proc. of BCS HCI 2007*, 205-213.
- Ramel, J.-Y., Vincent, N., & Emptoz, H. (1998). Interprétation de documents techniques par « cycles perceptifs » à partir d'une perception globale du document. *Revue Traitement du Signal*, 15(2), 1-20.
- Safin, S. (2011). *Processus d'externalisation graphique dans les activités cognitives complexes : le cas de l'esquisse numérique en conception architecturale individuelle et collective*. PhD Thesis, University of Liège, Belgium.
- Safin, S., Pecceu, M., & Leclercq, P. (2007). Interface-croquis pour relevés architecturaux: vers la constitution d'une nouvelle activité. *Proc. of IHM '07*.
- Shilman, M., & Viola, P. (2004). Spatial Recognition and Grouping of Text and Graphics. *Proc. of Eurographics SBIM 2004*, 91-95.
- Wuersch, M., & Egenhofer, M.J. (2008). Perceptual Sketch Interpretation. *Proc. of the 13th International Symposium on Spatial Data Handling (SDH 2008)*.