

# Vers un outil d'inspection temps réel de l'état d'avancement d'un chantier de construction par RA

M. Baubriaud<sup>1,2</sup>

S. Derrode<sup>1</sup>

R. Chalon<sup>1</sup>

K. Kernn<sup>2</sup>

<sup>1</sup> LIRIS, CNRS, UMR5205, Centrale Lyon, F-69130 Ecully, France

<sup>2</sup> SPIE Building Solutions, F-69320 Feyzin, France

{mathis.baubriaud, stephane.derrode, rene.chalon}@ec-lyon.fr

## Résumé

*Parmi les technologies clés de l'industrie 4.0, la Réalité Augmentée (RA) et le Building Information Modeling (BIM) sont deux des technologies les plus prometteuses pour assister les experts à inspecter un chantier de construction. Pour cette tâche, la pratique actuelle reste chronophage, laborieuse et manuelle, nécessitant une extraction exhaustive des données à partir de dessins et d'autres bases de données. Les systèmes basés sur des appareils mobiles pourraient accélérer et faciliter le processus d'inspection. À cette fin, un nouvel outil de RA pour assister les opérateurs lors des activités d'inspection de chantier en intérieur est proposé. La méthode introduite détecte automatiquement si les équipements du génie climatique et électrique (gainés, ventilation, chauffage, etc.) existent en réalité, conformément aux maquettes numériques 3D. Des expérimentations sur le terrain ont été menées afin d'estimer l'ensemble optimal de paramètres de la méthode et d'évaluer sa convivialité.*

## Mots clés

Réalité Augmentée, suivi de progression, BIM, inspection.

## 1 Introduction

Le suivi d'avancement des chantiers de construction est essentiel, car il donne aux gestionnaires les informations nécessaires pour agir rapidement et judicieusement [1]. Un suivi d'avancement inefficace entraîne une perte de contrôle du chantier, occasionnant des dépassements en temps et en coûts. La saisie manuelle des données requise par les techniques conventionnelles de supervision des progrès est laborieuse, chronophage et sujette à l'erreur humaine [2]. Les inspections des travaux intérieurs - par exemple, l'installation de chauffage, de ventilation et de climatisation (CVC) - peuvent s'avérer encore plus difficiles pour les inspecteurs en raison du niveau de détail et de l'interdépendance des tâches [3]. Par conséquent, il devient souhaitable d'automatiser ces tâches.

Le BIM est un ensemble de politiques, de processus et de technologies en interaction ayant pour but de gérer les

données essentielles d'un chantier, sous forme numérique, tout au long du cycle de vie de celui-ci [4]. Une étape cruciale pour se rapprocher de cet idéal réside dans la numérisation des maquettes en 3D.

La RA est l'une des techniques les plus prometteuses pour améliorer le transfert d'informations du monde numérique vers le monde physique de manière non intrusive [5]. La RA peut assister les opérateurs sur leur lieu de travail, en temps réel et pendant les opérations manuelles, en réduisant les erreurs humaines et la dépendance à la mémoire de l'opérateur, car elle a la capacité d'intégrer des informations virtuelles dans la perception du monde réel par l'utilisateur.

Des efforts ont été déployés vers des systèmes pour automatiser le suivi des progrès à l'aide de scanners laser, de photogrammétrie, de vidéogrammétrie et d'identification par radiofréquence (RFID) en lien avec des plateformes BIM [6]. De plus, certaines études ont tenté d'appliquer la RA au domaine de l'ingénierie architecturale [7, 8]. Cependant, la plupart de ces systèmes sont statiques, limités aux chantiers de construction extérieurs ou à des équipements spécifiques et ne fonctionnent pas en temps réel.

Cet article présente une méthode basée sur un appareil mobile pour la surveillance automatisée des progrès en temps réel d'un chantier. Les sections suivantes comprennent la mise en œuvre, les tests et les résultats de la méthode imaginée. Pour terminer, une discussion sur les limites et les futurs efforts de recherche est proposée.

## 2 Mise en œuvre de l'outil

La majorité des appareils qui prennent en charge les applications de RA sont des appareils RGBD, qui fournissent une reconstruction 3D de l'environnement. Les lunettes Microsoft HoloLens 2<sup>1</sup> embarquent deux capteurs de profondeurs, un accéléromètre, et un gyroscope, exploités pour fournir une cartographie spatiale de l'environnement. Il s'agit d'un appareil portable qui ne limite pas l'utilisation des mains, ce qui réduit les risques pour la sécurité et offre une expérience interactive et intuitive pour les utilisateurs.

1. <https://www.microsoft.com/fr-fr/hololens/hardware>

Cet article présente une méthode de comparaison automatisée et en temps réel entre le volume reconstruit capturé à partir de Hololens et la maquette 3D tel que planifiée. L'analyse est pensée pour fonctionner pendant que l'inspecteur navigue dans le bâtiment en construction. Une vue d'ensemble de la solution est présentée dans la Fig. 1

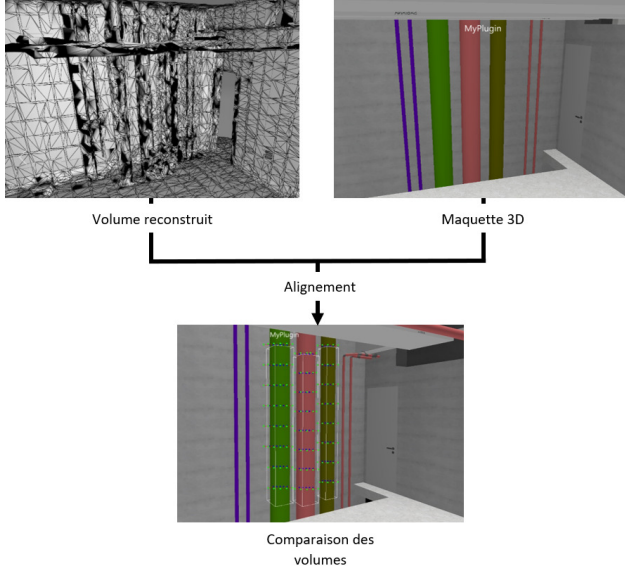


FIGURE 1 – Vue d'ensemble de la solution.

Le processus se compose de deux étapes principales, décrites dans les sous-sections suivantes. La première est un contrôle de visibilité des objets depuis le point de vue de la caméra dans la maquette 3D et la seconde permet la comparaison entre le volume reconstruit et les objets 3D considérés visibles. Le résultat final est la classification des objets 3D comme « construits » ou « non construits ». Ce travail s'inscrit dans une dynamique d'amélioration de l'application NEXT-BIM<sup>2</sup>. Ce logiciel permet la visualisation des maquettes 3D sur les lunettes ainsi que l'alignement semi-automatique du volume reconstruit.

## 2.1 Contrôle de visibilité des objets 3D

La première étape consiste à effectuer un contrôle de visibilité. Cette étape détecte quels objets de la maquette 3D sont visibles depuis la caméra de l'appareil mobile. La comparaison est effectuée pour les éléments visibles au fur et à mesure du déplacement de la caméra. L'emplacement de la caméra, son orientation et ses caractéristiques intrinsèques dans le monde et dans la maquette 3D sont récupérés automatiquement depuis les différents capteurs des lunettes. Un objet doit satisfaire quatre conditions pour être défini comme visible :

- L'objet est du type cible que nous cherchons à analyser;
- L'objet est dans le frustum de la caméra, qui correspond à la zone qui apparaît à l'écran;

- La distance entre l'objet et la caméra est inférieure à 5m, la qualité du maillage renvoyé par les lunettes devenant inexploitable au-delà;
- L'objet n'est pas occulté.

Pour chaque objet rendu dans la maquette 3D appartenant au type recherché, un simple calcul de la distance de celui-ci à la caméra permet de faire une présélection. Ensuite, plusieurs rayons, d'origines uniformément réparties dans l'objet et pointant vers la caméra, sont générés. Le nombre de ces rayons est proportionnel aux dimensions de l'objet. Si au moins 2 de ces rayons atteignent la caméra, l'objet est estimé visible. En outre, la position des points non masqués permet d'extraire uniquement la portion visible de l'objet 3D en question.

## 2.2 Analyse de présence d'un objet 3D

La maquette 3D ainsi que le volume reconstruit se compose de sommets reliés entre eux, formant un maillage triangulaire dense. Une description de l'algorithme élaboré pour l'analyse de présence est présentée dans l'Alg. 1. Dans un premier temps, une phase de prétraitement de la donnée est réalisée contenant 3 sous-étapes, illustrées dans la Fig. 2. Une projection des points 3D de l'objet sur l'écran 2D permet d'obtenir *projPts* 2D. L'enveloppe convexe 2D de *projPts* est calculée à l'aide d'une implémentation de la méthode du parcours de Graham [9]. Cette enveloppe représente maintenant la portion qu'occupe l'objet sur l'écran de l'utilisateur et l'Axis Aligned Bounding Box *aabb* est calculée pour simplifier la forme du polygone.

À l'aide des paramètres de cette surface 2D, de la taille maximale de l'objet 3D *obj\_size* et de la taille de l'écran *screen\_size*, un nombre de rayons *nRay[x, y]*, à générer le long de l'axe des abscisses *x* et des ordonnées *y* est défini puis ajusté<sup>3</sup> pour chaque objet à l'aide de la formule (1) :

$$nRay[x]* = obj\_size * \frac{\max(aabb[x]) - \min(aabb[x])}{screen\_size[x]} \quad (1)$$

Ces rayons, d'origines uniformément réparties dans l'enveloppe, sont projetés vers l'objet 3D. Cette méthode permet de récupérer un échantillon significatif de la forme 3D de l'objet tel qu'il est vu par l'utilisateur. Si un

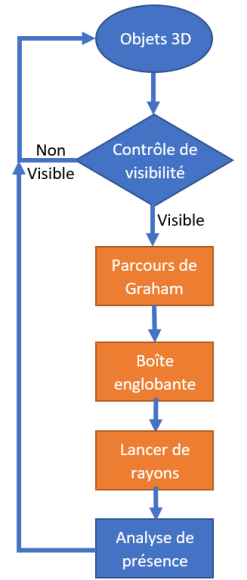


FIGURE 2 – Prétraitement à l'analyse de présence.

2. <https://next-bim.com/>

3. de manière équivalente pour l'axe *y*

rayon atteint bel et bien l'objet, la position 3D de l'impact ainsi que la normale à la surface de l'objet est enregistrée. Si le nombre de rayons atteignant l'objet dépasse un seuil *reachedRayThreshold* proportionnellement au nombre envoyé depuis l'enveloppe, alors l'analyse se poursuit, sinon l'objet est considéré de forme trop complexe pour pouvoir être étudié (au vu de la précision des capteurs Hololens). Pour chacun des points 3D enregistrés, un nouveau rayon ayant pour vecteur directeur la normale à l'objet et partant dans les deux directions avec une longueur *segmentSize* de 10 cm est créé. L'objectif est ainsi de détecter si une éventuelle intersection avec le volume reconstruit est identifiée et si la normale au point d'intersection est cohérente avec la forme de l'objet. Selon un seuil *intersectMeshThreshold* du rapport nombre d'intersections par le nombre de rayons générés, on considérera l'objet virtuel comme existant ou non existant dans la scène réelle.

---

**Algorithm 1** Analyse de présence.

---

```

1: Input : aabb, nRay, rayPositions
2: Output : presence
3: Data : nbrHits = 0, reachedRayThreshold = 0.25,
   segmentSize = 10
4: if size(rayPositions) > reachedRayThreshold
5: then
6:   for p in rayPositions do
7:     hit = GetRayIntersection(p.pos +
8:       segmentSize * p.norm, p.norm)
9:     if hit.HitDistance < segmentSize * 2 then
10:      //Il y a une intersection avec le maillage
11:      dans l'intervalle de 20cm
12:      nbrHits ++
13:    end if
14:  end for
15:  intersectMeshThreshold = 0.5
16:  if nbrHits > intersectMeshThreshold *
17:    size(rayPositions) then
18:    return presence = 1
19:  else
20:    return presence = 0
21:  end if
22: end if

```

---

Les seuils ont été déterminés par des expériences qui sont présentées dans la section suivante.

### 3 Expérimentation et résultats

L'environnement de développement de NEXT-BIM, le framework Microsoft Hololens et Microsoft Visual Studio 2022 ont permis la validation et les tests de la méthode proposée. L'algorithme est rédigé dans le langage de programmation C#. Des expériences ont été menées avec des données de différents chantiers de construction CVC, plomberie et électrique.

La solution implémentée sur les lunettes délivre une

performance de 10 à 30 images par seconde, inversement proportionnelle à la quantité d'objets d'intérêts à analyser dans une scène. Une étude qualitative de l'outil a été conduite, avec, comme critères, sa convivialité, sa pertinence et son acceptabilité. Un échantillon significatif de chantiers tertiaires de construction en intérieur fut sélectionné. Durant chaque visite, un expert sur place, externe au projet, testait la solution, et son ressenti était recueilli à l'aide d'un questionnaire. Différents points positifs et cas d'usages, mais également des voies d'améliorations en découlent.

L'outil apparaît aux premiers abords convaincants, les résultats sont immédiats et une aide visuelle à l'aide de couleur, verte ou rouge en fonction du bilan de l'analyse automatique permet de répertorier immédiatement la présence ou non des objets. Néanmoins, on observe rapidement des limitations matérielles importantes qui seront mentionnées dans la conclusion de cet article.

Dans l'optique de réduire le facteur humain et matériel, l'utilisation d'un outil développé par NEXT-BIM permet d'enregistrer les relevés des capteurs des lunettes. Ainsi, il est possible de rejouer une scène en temps réel, reproduisant les mouvements exacts de l'utilisateur dans le bâtiment et limitant les résultats d'analyse aux seules constantes de l'algorithme proposé. Un exemple de l'interface de cet outil est présenté en Fig. 3.

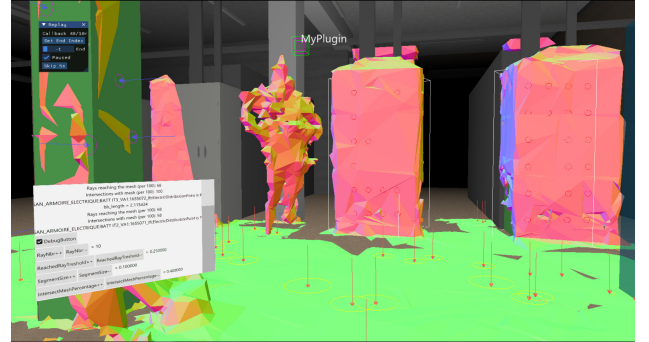


FIGURE 3 – Exemple d'interface d'expérimentation.

Les métriques pour évaluer les résultats sont l'exactitude, la précision et le rappel. L'exactitude indique à quel point l'algorithme est correct dans l'ensemble. La précision spécifie le taux d'éléments retrouvé rapporté au nombre d'éléments total et le rappel indique le taux d'éléments retrouvé rapporté au nombre d'éléments construits. La vérité terrain est définie en étiquetant manuellement tous les objets comme existant ou non. Une étude de l'impact des constantes de l'algorithme révèle que les deux constantes dont la sensibilité est la plus haute sont la taille des segments *segmentSize* qui vérifie l'intersection avec le maillage Hololens et le seuil *intersectMeshThreshold* nécessaire pour que l'objet soit considéré présent. Le Tabl. 1 présente les résultats obtenus dans le cas d'une pièce d'un datacenter en construction contenant 10 armoires élec-

triques dans la maquette 3D dont 5 correctement placés à la date de l'inspection. Malgré le nombre de données réduites

			Taille des segments		
			7	10	13
Seuil	25%	Exactitude	60	90	80
		Précision	<b>100</b>	83	71
		Rappel	60	<b>100</b>	<b>100</b>
	50%	Exactitude	70	<b>100</b>	90
		Précision	<b>100</b>	<b>100</b>	<b>100</b>
		Rappel	70	<b>100</b>	83
	75%	Exactitude	50	80	90
		Précision	<b>100</b>	<b>100</b>	<b>100</b>
		Rappel	40	60	83

TABLEAU 1 – Pourcentage d'exactitude, précision et rappel pour différents seuils et tailles de segments.

lors de ce test, on observe bien des tendances. Lorsque les segments sont trop courts, de moins nombreuses intersections sont relevées entre la maquette 3D et le volume reconstruit ; inversement des faux positifs sont observés lorsqu'ils sont trop longs. Les expériences et l'évaluation des résultats suggèrent que les paramètres optimaux de seuil et de taille de segment pour la solution proposée sont respectivement de 50% et 10 cm. Toutefois, ces valeurs ne peuvent pas être généralisées pour tous les types d'objets 3D. Des résultats différents pourraient être obtenus dans d'autres cas d'usage.

## 4 Conclusion et perspectives

Les pratiques actuelles d'inspection de chantier sont principalement manuelles et demandent une main-d'œuvre importante. Cet article présente une solution mobile de suivi des progrès automatisés en temps réel. Pendant que l'inspecteur navigue à l'intérieur d'un bâtiment en construction, l'outil développé analyse les objets de la maquette 3D et un algorithme basé sur les lancers de rayons permet de les comparer avec le volume reconstruit par les lunettes. Des expérimentations ont permis d'optimiser les paramètres de la solution. Néanmoins, de plus amples tests doivent être conduits pour affiner ses constantes, en particulier dans des scénarios différents et avec un plus grand nombre de données. La méthode actuelle est prometteuse, mais présente comme défaut principal de reposer uniquement sur des données de profondeurs, peu fiables et précises. De plus, il existe parfois de grosses disparités entre le BIM et le chantier à tel point que des décalages de plusieurs dizaines de centimètres peuvent être observés. Dans ces cas, un algorithme est en cours de conception pour calculer ces décalages. Également, l'usage d'un casque de RA dans un environnement complexe tel qu'un chantier soulève des questions de sécurité et de prise en main de la technologie. D'autres travaux sont en cours pour pallier ces manques en utilisant par exemple la caméra des lunettes. Bien que la solution proposée ait été développée pour Microsoft Hololens, elle pourrait s'appliquer à tout appareil

dont les données d'entrée et de sortie sont similaires (c'est-à-dire un maillage construit grâce au capteur de profondeur de temps de vol (ToF) des lunettes).

En conclusion, de nombreux cas d'usage ont été imaginés et le potentiel d'un tel outil est justifié. Par conséquent, des développements supplémentaires doivent être menés.

## Remerciements

La recherche menant à ces résultats a été financée par une thèse co-encadrée entre l'École Centrale Lyon et la société SPIE Building Solutions. L'entreprise NEXT-BIM est remerciée pour son soutien dans la mise en place de la solution dans leur environnement de travail et l'utilisation de leurs outils.

## Références

- [1] V. K. Reja, K. Varghese, et Q. P. Ha. Computer vision-based construction progress monitoring. *Automation in Construction*, 138 :104245, juin 2022.
- [2] Jochen Teizer. Status quo and open challenges in vision-based sensing and tracking of temporary resources on infrastructure construction sites. *Advanced Engineering Informatics*, 29(2) :225–238, avril 2015.
- [3] B. Koo et M. Fischer. Feasibility study of 4D CAD in commercial construction. *Journal of Construction Engineering and Management*, 126(4) :251–260, juillet 2000. Publisher : American Society of Civil Engineers.
- [4] Rafael Sacks, Charles Eastman, Ghang Lee, et Paul Teicholz. *BIM Handbook : A Guide to Building Information Modeling for Owners, Designers, Engineers, Contractors, and Facility Managers*. Wiley, août 2018.
- [5] F. De Pace, F. Manuri, et A. Sanna. Augmented Reality in Industry 4.0. *American Journal of Computer Science and Information Technology*, 06(01), 2018.
- [6] T. Omar et M. L. Nehdi. Data acquisition technologies for construction progress tracking. *Automation in Construction*, 70 :143–155, octobre 2016.
- [7] C. Park, D. Lee, O. Kwon, et X. Wang. A framework for proactive construction defect management using BIM, augmented reality and ontology-based data collection template. *Automation in Construction*, 33 :61–71, août 2013.
- [8] M. Kopsida et I. Brilakis. Real-time volume-to-plane comparison for mixed reality-based progress monitoring. *Journal of Computing in Civil Engineering*, 34(4) :04020016, juillet 2020.
- [9] X. Kong, H. Everett, et G. Toussaint. The Graham scan triangulates simple polygons. *Pattern Recognition Letters*, 11(11) :713–716, novembre 1990.