



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

المدرسة الوطنية العليا للتكنولوجيات المتقدمة
Ecole Nationale Supérieure des Technologies Avancées



Département Génie Électrique et Informatique Industrielle

Mémoire de fin d'étude en vue de l'obtention du diplôme Master

Filière :
Automatique

Spécialité :
Automatique et Informatique Industrielle

Thème :

Détection de la qualité de peinture des extincteurs à
l'aide d'une caméra Kinect et de techniques de
traitement d'image

Réalisé par :

NEDJAR Ahmed Amine & GHERAB Aimen

Soutenu le 23 / 06 / 2025

Membres du jury :

Nom et Prénom	Établissement	Grade	Qualité
Mme. BOUTARFA Souhila	ENSTA	M.C.B	Président
M. DERMOUCHE Reda	ENSTA	M.C.A	Promoteur
M. BAMMOUNE Fayçal	FB Solution	Expert métier	Co-Promoteur
M. GUELLAL Amar	ENSTA	M.C.B	Examineur
M. BEGHAMI Sami	ENSTA	M.A.A	Examineur

Année universitaire : 2024/2025

Détection de la qualité de peinture des extincteurs à l'aide d'une caméra Kinect et de techniques de traitement d'image

GHERAB Aimen

*Département de Génie Électrique
et Informatique Industrielle
École Nationale Supérieure des
Technologies Avancées
Alger, Algérie
a_gherab@ensta.edu.dz*

NEDJAR Ahmed Amine

*Département de Génie Électrique
et Informatique Industrielle
École Nationale Supérieure des
Technologies Avancées
Alger, Algérie
a_nedjar@ensta.edu.dz*

Résumé – Ce mémoire présente la conception et la mise en œuvre d'un système automatique de contrôle de la qualité de peinture appliquée sur des extincteurs. Le dispositif utilise une **caméra Kinect Xbox 360** couplée à un programme développé sous **MATLAB** pour effectuer l'analyse d'images. Le cœur de la méthode repose sur la *segmentation colorimétrique*, le *suivi de mouvement* par l'algorithme de *Lucas-Kanade*, l'extraction de la *région d'intérêt (ROI)*, et le calcul de la *composante rouge moyenne*. Des seuils de décision permettent ensuite de classer les extincteurs selon leur qualité de peinture. Les résultats expérimentaux obtenus montrent une bonne précision et valident la pertinence d'un tel système dans un contexte industriel automatisé.

Mots-clés – Vision industrielle, qualité de peinture, caméra Kinect Xbox 360, MATLAB, traitement d'image, suivi de mouvement, Lucas-Kanade, segmentation, ROI, inspection automatique.

1 Introduction

La qualité des processus industriels est un défi crucial pour les industries qui veulent assurer la fiabilité et la réputation de leurs produits et réduire les coûts de fabrication. Dans ce contexte, la vision industrielle est un point focal, en particulier pour la détection de qualité de production, où la vitesse et la précision sont essentielles.[1]

La vision industrielle est une technologie qui permet aux machines de "voir" en utilisant des caméras et des algorithmes de traitement d'image pour analyser automatiquement des objets ou des scènes. Elle est largement utilisée dans les processus industriels pour réaliser des tâches comme l'inspection qualité, le guidage de robots, la mesure ou encore la reconnaissance de formes. Grâce à sa rapidité, sa précision et sa capacité à fonctionner sans interruption, elle remplace avantageusement le contrôle visuel humain, en garantissant une meilleure répétabilité et en réduisant les erreurs liées à la fatigue ou à l'interprétation subjective.[2]

La technique mise en œuvre repose sur quelques étapes de base à l'aide d'une caméra KINECT XBOX 360 : la détection de la forme de la bouteille d'extincteur, le suivi de ses contours à l'aide de Lucas-Kanade, suivie de l'extraction

de la région d'intérêt d'analyser enfin la qualité de la peinture en calcul de la moyenne de la couleur rouge (composante R du modèle RGB). Cette technique permet de déterminer automatiquement si la surface est bien peinte ou non, en fonction de l'utilisation de l'environnement de développement MATLAB.[3]

Notre travail présente d'abord le contexte industriel de la chaîne de peinture, puis esquisse la méthodologie de traitement d'image appliquée à l'auto-inspection visuelle matée, avant de discuter des résultats obtenus et les perspectives d'amélioration du système .

1.1 Contexte industriel

Au sein du secteur industriel, l'ultime contrôle que constitue la qualité est une opération fondamentale pour assurer la conformité et la fiabilité des produits à livrer. Cela est particulièrement vrai dans certains secteurs très spécifiques comme celui de l'industrie de peinture, notamment pour l'automobile, l'électroménager ou la fabrication d'extincteurs, où l'application correcte de la peinture conditionne, non seulement l'aspect, mais aussi la protection contre la corrosion et la durée de vie du produit. Une irrégularité sur la surface ou une zone non peinte entraîne le re-

fus de la pièce ou un retour client (produit non conforme), responsable d'une augmentation des coûts de production et d'une image dégradée de l'entreprise.[4]

1.2 Problématique du contrôle visuel manuel

Aujourd'hui encore, de nombreuses entreprises ont recours à une inspection manuelle du revêtement de peinture. Cette méthode repose sur la vigilance et l'expérience de l'opérateur. Or, elle présente plusieurs limitations : variabilité des jugements, fatigue visuelle, manque de reproductibilité, et impossibilité d'assurer un contrôle continu à haute cadence . Ces failles deviennent d'autant plus critiques à mesure que les lignes de production sont automatisées et rapides.[5]

Dans ce contexte, l'inspection manuelle ne répond plus aux exigences de performance des environnements industriels modernes. Elle engendre des risques accrus d'erreurs de jugement, de retards dans la détection des défauts et d'augmentation des coûts liés aux reprises ou aux rejets. C'est pourquoi l'automatisation du contrôle qualité par des systèmes de vision industrielle devient non seulement une solution efficace, mais aussi une nécessité pour garantir une production fiable, rapide et conforme aux standards actuels.

1.3 Intérêt d'une automatisation par vision

Pour remédier à ces inconvénients, la radicalité de l'automatisation par présence de vision artificielle donne de grands résultats. Par l'utilisation d'une caméra associée avec un algorithme de traitement d'image, on parvient facilement à réaliser la détection des défauts de peinture tels que les problèmes de teinte, sur épaisseurs, et coulures par un contrôle à la fois rapide, objectif et répétitif, diminuant la dépendance à l'humain.

Elle trouve tout son intérêt dans cette logique d'industrie 4.0 confrontée à l'essor de la supervision intelligente des procédés.[6]

1.4 Objectif du projet

Dans ce contexte, le présent travail vise à concevoir un système automatique permettant de contrôler la qualité de la peinture appliquée sur

des extincteurs. Le système repose sur une caméra Xbox 360 Kinect, capable de capturer à la fois l'image couleur et la profondeur de la scène. L'analyse des données est assurée par un programme développé sous MATLAB, qui détecte d'abord la forme de l'extincteur, extrait la zone peinte, puis mesure la moyenne de la composante rouge (RGB) afin d'évaluer la conformité de la peinture. Ce système se veut simple, économique et adaptable, notamment pour les structures industrielles de petite et moyenne taille

2 Matériel utilisé

Cette section décrit les équipements matériels et logiciels nécessaires à la mise en uvre du système de détection automatique, incluant la caméra, l'environnement de développement, ainsi que les conditions expérimentales d'acquisition des données.

2.1 Description de la caméra Kinect Xbox 360

Le capteur Microsoft Kinect est une caméra de balayage 3D à faible coût, haute résolution et courte portée développé pour le système de jeu vidéo Microsoft Xbox 360. Le Kinect, comme indiqué à la figure 1.



FIGURE 1 – Caméra Kinect Xbox 360 [7]

présente une variété d'avantages tels qu'il peut mesurer la profondeur et la couleur à la vitesse vidéo simultanément dans comparaison avec un autre type de scanner 3D. Le scanner Kinect est un concept basé sur un périphérique complémentaire de type webcam. Kinect contient trois éléments essentiels qui ensemble pour détecter le mouvement des utilisateurs et créer l'image physique des utilisateurs sur l'écran : un RGB (rouge, vert, bleu) avec un capteur de profondeur (D),

une caméra VGA (vidéo graphique) et un multi-réseau microphone . Ces spécifications du Kinect fournissent un critère vital en tant que grand scanner pour obtenir la topographie du fantôme.

La Kinect Xbox 360 joue un rôle central dans notre système de contrôle automatique de la qualité de peinture. Son capteur RGB permet d'analyser précisément la teinte rouge de la peinture appliquée, critère essentiel pour juger la qualité du revêtement. Le capteur de profondeur facilite la détection et l'isolation de la forme de l'extincteur, améliorant la précision du traitement d'image. La caméra VGA, avec une résolution adaptée, assure un suivi fluide de l'objet en mouvement, contribuant ainsi à la fiabilité du système de détection.

2.2 Brève fiche technique

Le tableau 1 présente une fiche technique simplifiée de la caméra Kinect Xbox 360 utilisée dans le cadre de ce projet, en mettant en évidence ses principales caractéristiques matérielles.

Composant	Caractéristiques
Capteur RGB	Résolution : 640 (E 480 px à 30 fps. Utilisé pour l'analyse des couleurs (teinte rouge).
Capteur de profondeur	Résolution : 320 (E 240 px à 30 fps. Portée : 0,8 m à 4 m. Sert à détecter la forme 3D de l'extincteur.
Champ de vision	Horizontal : 57° Vertical : 43°
Connexion	USB 2.0 (alimentation externe requise sur PC).

TABEAU 1 – Fiche technique simplifiée de la caméra Kinect Xbox 360

2.3 Environnement de développement : MATLAB

Dans le cadre du développement, retenu pour la solution de détection automatique de la qualité de peinture, le choix s'est porté sur l'environnement de développement MATLAB (Matrix Laboratory). MATLAB est un langage de programmation de haut niveau associé à un environnement

intégré, particulièrement adapté pour le traitement d'image, la vision par ordinateur, et l'analyse de données. De plus, il possède de nombreuses boîtes à outils (Toolbox), comme la Computer Vision Toolbox et la Image Processing Toolbox, qui sont pourtant essentielles au développement du projet.

En effet, ces bibliothèques permettent la détection d'objets ainsi que le traitement des couleurs (comme le modèle RGB, le suivi de mouvement (comme l'algorithme de Lucas-Kanade, l'extraction de caractéristiques visuelles, tout en assurant une visualisation rapide et efficace des résultats, accélérant ainsi le processus de test et de validation des algorithmes).

Ainsi, la convivialité de l'interface, les fonctions intégrées particulièrement puissantes et la documentation riche de MATLAB en font un outil de choix pour le prototypage rapide et les projets académiques en vision industrielle.[3]

2.4 Conditions de test

Les différentes épreuves du système de contrôle automatique de la qualité de peinture ont eu lieu dans des conditions simulant le réel d'une chaîne de peinture industrielle, sur un banc d'essai composé des éléments suivants :

- **Extincteurs métalliques peints** : des pièces de différentes teintes rouges, certaines volontairement mal peintes (zones plus claires ou non couvertes) afin d'établir la capacité de détection du système.
- **Caméra Kinect Xbox 360** : placée à environ 1 à 1,2 m des objets, dans la plage de distance optimale pour l'évaluation de la profondeur.
- **Éclairages** : les tests ont été réalisés dans des conditions d'éclairages stables pour éviter les reflets directs pouvant altérer les couleurs capturées par la caméra RGB.
- **Support mobile ou fixe** les extincteurs étaient soit fixés, soit déplacés lentement pour simuler un transit sur un convoyeur. Le traitement a eu lieu sur un ordinateur portable équipé de MATLAB R2023b, en configuration standard.

L'algorithme a été testé en quasi temps réel sur des séquences vidéo ou des images prises individuellement par la Kinect. Ces essais ont permis d'évaluer la robustesse du système dans un contexte semi-contrôlé, en vue d'un futur déploiement.

ment en milieu industriel réel. Les différentes conditions expérimentales dans lesquelles les tests ont été réalisés sont résumées dans le tableau 2. Elles permettent de simuler un environnement proche de celui d'une chaîne industrielle.

Paramètre	Valeur / Condition
Type de caméra	Kinect Xbox 360
Hauteur de la caméra	1,2 m
Position de la caméra	Fixe, perpendiculaire à l'objet
Type d'objet	Extincteurs industriels
Éclairage ambiant	Fluorescent blanc, diffus
Contexte de test	Objet immobile et en mouvement
Logiciel utilisé	MATLAB R2023a
Type de traitement	Image RGB et profondeur
Support de déplacement	Plateforme motorisée simulant un convoyeur

TABLEAU 2 – Résumé des conditions de test

3 Méthodologie

La méthodologie consiste à analyser la composante rouge (R) d'images extraites d'une vidéo filmée par une caméra Kinect Xbox 360, après extraction de la région d'intérêt correspondant à la zone peinte de l'extincteur. La moyenne des valeurs rouges est calculée et comparée à un seuil, permettant de détecter automatiquement les défauts de peinture de manière simple et rapide. [8]

3.1 Détection de la forme de l'extincteur

La détection de la forme de l'extincteur constitue la première étape clé du traitement d'image. Elle vise à localiser précisément l'objet à inspecter dans l'image acquise par la caméra Kinect. Cette étape repose sur deux sous-procédés : **la segmentation couleur suivie d'une extraction des contours binairese** .

3.1.1 Segmentation par couleur (création du masque binaire)

La technique de traitement d'image appelée de " segmentation de couleur " permet de regrou-

per les pixels d'une image en des groupes homogènes selon leur couleur à la fin d'isoler des objets ou des zones d'intérêt en fonction des coordonnées chromatiques (telles que RGB ou HSV) du pixel. En profitant de la disparité de teinte, saturation ou d'intensité lumineuse entre un objet à détecter et le milieu ambiant, des objets d'un plan d'image peuvent être identifiés d'après leurs propriétés chromatiques. Cette méthode de traitement de l'image constitue de l'importance majeure dans la situation où la teinte d'un objet à identifier serait dissimilante de celle du fond.

Gonzalez et Woods observe que la technique de segmentation par couleur se base sur l'hypothèse que l'objet d'intérêt peut s'distinguer du fond sous le critère chromatique caractéristique à lui-même. Elle correspond à un style de fonctionnement commun de système de vision à utiliser pour suivi, détection, et analyse de l'objet dans environnement contrôlée.[9]

Algorithme général de segmentation de couleur :

Le processus typique d'un algorithme de segmentation de couleur suit plusieurs étapes clés :

- **Acquisition de l'image couleur** : il s'agit de récupérer une image RGB à partir d'un capteur ou d'une caméra.
- **Conversion de l'espace couleur** : dans certains cas, l'image peut être convertie de l'espace RGB vers un autre espace couleur plus adapté à la perception humaine, comme HSV, Lab ou YCbCr.
- **Extraction des canaux pertinents** : les canaux les plus représentatifs sont sélectionnés selon la teinte recherchée. Dans le cas de la détection d'un extincteur rouge, le canal rouge est privilégié.
- **Application d'un seuillage** : cette étape permet de segmenter l'image en distinguant les pixels d'intérêt (ayant une forte intensité dans le canal cible) des autres, générant ainsi une image binaire.
- **Filtrage morphologique (optionnel)** : des opérations comme l'érosion, la dilata-tion ou l'ouverture peuvent être appliquées afin d'éliminer les bruits et de nettoyer le masque binaire.

Le résultat final est un masque binaire qui met en évidence la zone d'intérêt, ici caractérisée par une couleur rouge dominante. Cette étape est cru-

ciale pour toute analyse ultérieure de forme ou de texture.

Dans le cadre de notre projet, la segmentation par couleur a été utilisée pour isoler automatiquement les extincteurs peints en rouge à la sortie de la chaîne de peinture. L'image capturée par la caméra Xbox 360 est exploitée en mode RGB sans conversion, car la teinte rouge y est déjà bien distincte [10].

L'algorithme mis en œuvre suit les étapes suivantes :

- **Extraction du canal rouge (R)** de l'image RGB.
 - **Application d'un seuillage empirique** pour générer une image binaire dans laquelle seules les zones fortement colorées en rouge sont conservées.
- Les conditions suivantes ont été utilisées pour isoler efficacement le rouge :
- $R > 0,4$: pour écarter les zones trop sombres (valeurs RGB normalisées entre 0 et 1).
 - $R > G + 0,1$ et $R > B + 0,1$: pour assurer que le rouge est dominant par rapport aux autres composantes.

Les figure 2 et figure 3 illustrent respectivement l'image RGB d'entrée et le masque binaire obtenu après détection de la couleur rouge.

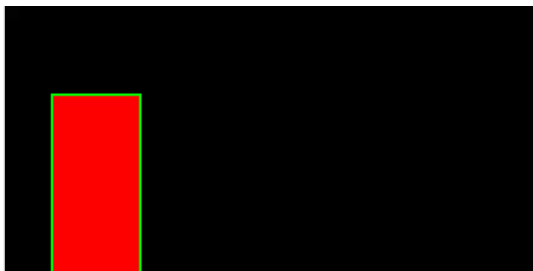


FIGURE 2 – Image d'entrée en mode RGB

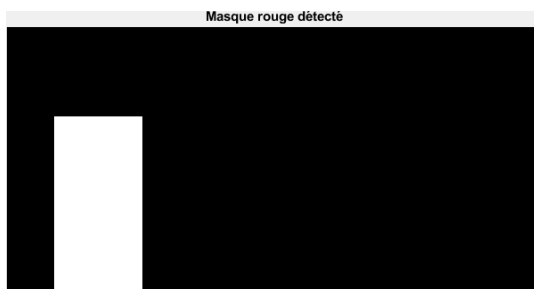


FIGURE 3 – Masque binaire obtenu après segmentation du rouge

3.1.2 Post-traitement morphologiques

Le post-traitement morphologique représente un ensemble d'opérations que l'on applique en traitement d'image binaire dans le but de corriger, d'améliorer ou de simplifier les formes repérées dans sa région d'intérêt après une étape de segmentation. L'ensemble des techniques mises en œuvre provient d'un cadre théorique, la morphologie mathématique, qui a été développée pour analyser la géométrie des objets présents dans des images binaires [11].

Les opérations morphologiques permettent notamment de :

- Supprimer les petits bruits ou petits objets isolés.
- Boucher les trous présents à l'intérieur des objets.
- Lisser les contours.
- Séparer ou de reconnecter des parties d'un même objet.

Algorithmes morphologiques courants

Les opérations de morphologie mathématique appliquées aux images binaires permettent de modifier les formes des objets présents dans une image selon des règles simples mais puissantes. Les principales opérations sont les suivantes :

- **Érosion** : cette opération rétrécit les objets en supprimant les pixels situés sur leurs contours. Elle est utile pour éliminer le bruit ou séparer des objets proches.
- **Dilatation** : elle agrandit les objets en ajoutant des pixels autour de leurs contours. Elle permet de renforcer la structure des objets et de combler des petites discontinuités.
- **Ouverture** : combinaison d'une érosion suivie d'une dilatation. Elle permet de supprimer les petits objets ou imperfections tout en préservant la forme générale des objets principaux.
- **Fermeture** : combinaison d'une dilatation suivie d'une érosion. Elle permet de boucher les petits trous ou fentes dans les objets binaires.

Toutes ces opérations reposent sur l'utilisation d'un *élément structurant* souvent une matrice carrée, un disque ou un autre motif géométrique qui détermine la manière dont les pixels sont ajoutés ou supprimés lors des traitements [12].

La figure 4 illustre les effets visuels typiques de ces opérations sur une image binaire.

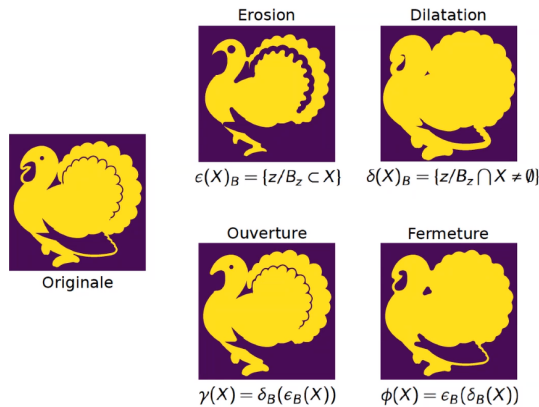


FIGURE 4 – Illustration des principales opérations morphologiques : érosion, dilatation, ouverture et fermeture [12]

Dans cette démarche, les opérations morphologiques ont permis de nettoyer l'image binaire issue de la segmentation par couleur afin d'obtenir une forme de l'extincteur propre et claire (sans artefacts) pour extraire les contours et évaluer correctement le niveau de qualité de peinture.

Les étapes de post-traitement appliquées dans MATLAB sont les suivantes :

- **Remplissage de trous (imfill)** pour supprimer les petits objets parasites (taches rouges non pertinentes).
- **Fermeture morphologique (imclose)** pour combler les petits trous ou irrégularités dans le sein de l'objet segmenté.
- **Suppression des petits objets isolés (bwareaopen)** dont la taille (exprimée en pixels) est inférieure à un certain seuil afin de ne conserver que l'objet principal de la scène d'image : l'extincteur.

Ces opérations ont permis d'obtenir une image binaire propre et fidèlement conforme à la morphologie réelle de l'objet peint ce qui est bien nécessaire pour les phases suivantes de détecteur de contours et d'études colorimétriques.

3.1.3 Extraction des contours binaires

La détection des contours constitue une étape essentielle dans le traitement d'images. Elle vise à localiser les limites d'un objet ou d'une région dans une image, c'est-à-dire les zones où se produisent des variations brusques en intensité (niveaux de gris) ou en couleur. Ces contours sont souvent exploités dans des tâches de reconnais-

sance d'objets, de suivi, de segmentation, de mesure ou encore de classification.

Parmi les nombreuses méthodes disponibles, le détecteur de Canny est l'un des plus réputés pour sa performance. Selon John Canny, un bon détecteur de contours doit satisfaire trois critères fondamentaux :

- Réduction efficace du bruit ;
- Localisation précise des bords ;
- Une seule réponse par contour détecté.

La figure 5 illustre un exemple typique de détection des contours binaires appliquée à une image.

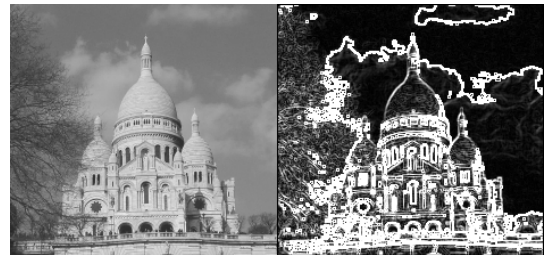


FIGURE 5 – Exemple de détection de contours sur une image binaire selon la méthode de Canny [13]

Algorithme général d'extraction de contours

Différentes techniques permettent d'extraire les contours d'une image binaire ou en niveaux de gris, parmi lesquelles les plus connues sont :

- **Le détecteur de Sobel**, qui est un détecteur de dérivée spatiale appliqué au traitement d'image par convolution, ce qui le rend simple et rapide.
- **Le détecteur de Prewitt** qui est une version peu pondérée du détecteur de Sobel.
- **Le détecteur de Canny** qui est une méthode plus complexe basée sur du filtrage gaussien suivi d'un calcul de gradient puis d'une suppression non maximales puis d'hystérésis (très précis, souvent utilisé au sein de l'industrie).

- **Méthode des contours de région**, qui permet d'extraire les contours dans une image binaire en effectuant la détection de transition allant de 0 à 1 à la limite de séparation des objets, le plus souvent via la fonction bwboundaries ou edge (MATLAB).[9]

Dans le cadre de notre système de contrôle automatisé de la qualité de peinture, l'extraction des contours joue un rôle essentiel pour localiser avec

précision la zone peinte de l'extincteur et pour quantifier la couverture.

Après la segmentation par couleur et le nettoyage morphologique de l'image binaire, l'extincteur apparaît clairement comme un objet blanc (valeur 1) sur fond noir (valeur 0). L'extraction des contours est alors utilisée pour détecter les limites précises de cette région binaire, ce qui permet :

1. De vérifier la forme globale de l'objet peint, et ainsi identifier toute déformation ou irrégularité dans la peinture.
2. D'extraire une région d'intérêt (ROI) sur l'image d'origine, en se basant sur les coordonnées des contours. Cela permet de limiter l'analyse colorimétrique uniquement à la zone peinte, et d'éviter les biais dus à l'arrière-plan.
3. D'analyser la régularité des contours, indicateur important d'un dépôt homogène de la peinture. Des irrégularités peuvent révéler une mauvaise application, une buse obstruée ou une instabilité du convoyeur.
4. D'aligner les images automatiquement lors de l'analyse série (plusieurs extincteurs), car les contours peuvent servir de repères géométriques pour normaliser les positions.

L'algorithme utilisé pour extraire les contours dans **MATLAB** repose sur la fonction **bwboundaries**, laquelle retourne les coordonnées des pixels situés sur la frontière des objets binaires détectés. Ces coordonnées sont ensuite exploitées dans deux objectifs principaux :

- Tracer la silhouette de l'objet (ici, un extincteur) directement sur l'image d'origine, ce qui permet une superposition visuelle intuitive ;
- Générer un masque précis délimitant uniquement la zone d'intérêt, facilitant ainsi les analyses ultérieures (forme, couleur, position, etc.).

La figure 6 illustre le résultat de cette extraction, où le contour de l'extincteur est mis en évidence en vert sur l'image segmentée.

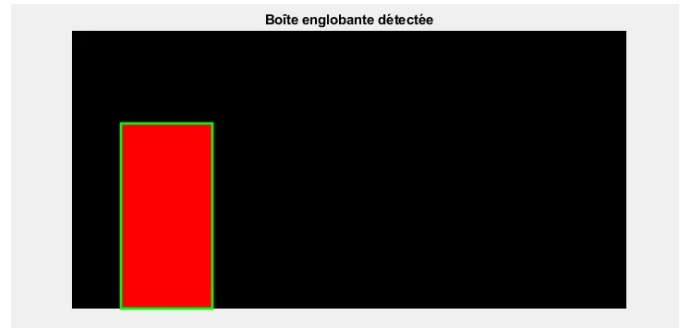


FIGURE 6 – Contour extrait de l'extincteur segmenté à l'aide de la fonction **bwboundaries** dans MATLAB.

Ainsi, l'extraction des contours ne constitue pas une étape purement visuelle ou esthétique, mais représente une condition essentielle à la fiabilité de l'analyse colorimétrique. Sans une délimitation précise de la région d'intérêt, il serait difficile de garantir que la teinte mesurée provient uniquement de la surface peinte de l'extincteur, et non d'éléments parasites du fond ou d'autres objets présents dans l'image.

La figure 7 présente un exemple concret issu d'une séquence réelle, où le contour de l'extincteur a été détecté de manière efficace.



FIGURE 7 – Exemple réel de détection de contours de l'extincteur à l'aide de la segmentation binaire.

3.2 Suivi du mouvement Algorithme Lucas-Kanade

L'algorithme de Lucas-Kanade permet de suivre le déplacement d'un objet à travers une séquence d'images en estimant localement le flux optique. Il est utilisé ici pour assurer le suivi précis de la région d'intérêt (ROI) représentant l'extincteur, tout en garantissant la cohérence spatiale des mesures de couleur entre les différentes images de la vidéo.

3.2.1 Introduction au suivi de mouvement

Le suivi de mouvement représente un problème central de la vision par ordinateur, notamment en ce qui concerne le suivi d'objets d'intérêt d'une image à l'autre dans une séquence vidéo. Pour notre projet, nous l'appliquons aux extincteurs se déplaçant le long d'une chaîne de peinture automatisée de manière à faire en sorte que la zone analysée pour la qualité de la peinture soit bien positionnée pendant tout mouvement de l'objet avec sa position réelle.

À cet effet, nous avons décidé d'utiliser l'algorithme de Lucas-Kanade, qui est à la fois simple, rapide et robuste dans des environnements industriels. [14]

3.2.2 Présentation de l'algorithme

1. **Objectif du suivi de mouvement** Le suivi de mouvement a pour objectif de détecter dans une séquence d'images le déplacement d'un ou plusieurs objets à la fois dans le temps. C'est une étape essentielle dans de nombreuses applications : vidéosurveillance, robotique, analyse biomédicale, etc. Il s'agit de localiser les objets dans chaque image et de retrouver leurs trajets, malgré les changements en position, en forme ou en éclairage.
2. **Principe du flux optique** Le suivi d'un objet dans une séquence vidéo repose sur la notion de flux optique (optical flow), qui décrit la manière dont l'intensité des pixels change entre deux images consécutives.

L'hypothèse principale est que **l'intensité d'un pixel reste constante entre deux images proches dans le temps** :

$$I(x, y, t) = I(x + u, y + v, t + 1) \quad (1)$$

- $I(x, y, t)$ est l'intensité du pixel à la position (x, y) au temps t ,
- u et v sont les composantes du mouvement dans les directions horizontale et verticale.

En développant cette équation en série de Taylor, on obtient l'équation du flux optique :

$$I_x \cdot u + I_y \cdot v + I_t = 0 \quad (2)$$

avec :

- I_x, I_y : dérivées partielles de l'image selon x et y ,
- I_t : dérivée temporelle (variation entre deux images),
- u, v : vecteurs du mouvement à estimer.

3. **Méthode de résolution** Ce système est indéterminé (une équation pour deux inconnues). Lucas et Kanade[14] ont proposé de le résoudre localement, dans une petite fenêtre \mathbf{W} fenêtre de $(3 \times 3$ ou $5 \times 5)$ autour du point considéré, en supposant que tous les pixels de cette fenêtre ont le même vecteur de mouvement.

On obtient **un système surdéterminé**.

Le système d'équations du flux optique s'écrit :

$$\begin{cases} I_x(p_1)u + I_y(p_1)v = -I_t(p_1) \\ I_x(p_2)u + I_y(p_2)v = -I_t(p_2) \\ \vdots \\ I_x(p_n)u + I_y(p_n)v = -I_t(p_n) \end{cases} \quad (3)$$

Sous forme matricielle :

$$A \begin{bmatrix} u \\ v \end{bmatrix} = b$$

où :

$$A = \begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_n) & I_y(p_n) \end{bmatrix}, \quad b = \begin{bmatrix} -I_t(p_1) \\ -I_t(p_2) \\ \vdots \\ -I_t(p_n) \end{bmatrix}$$

La solution par **moindres carrés** donne :

$$\begin{bmatrix} u \\ v \end{bmatrix} = (A^T A)^{-1} A^T b \quad (4)$$

où A contient les gradients spatiaux et b contient les dérivées temporelles.[14]

3.2.3 Application au suivi de la forme détectée

Objectif du suivi

L'objectif dans notre projet est de **suivre la zone de l'extincteur analysée pour la qualité de la peinture** tout au long de la séquence vidéo. Cela permet :

- De gagner du temps de traitement (important pour une application temps réel).

- D'assurer que les mesures de couleur (teinte rouge) soient prises sur la même surface physique, malgré les petits déplacements.[15]

Étapes détaillées

1. Prétraitement

- Lecture de la première image de la vidéo.
- Conversion en niveaux de gris avec **rgb2gray** car l'algorithme Lucas-Kanade s'applique à l'intensité lumineuse.

2. Détection de l'objet (extincteur rouge)

- Nous avons utilisé un masque de couleur pour détecter le rouge, en HSV et RGB.
- Le masque est nettoyé morphologiquement (fermeture, remplissage des trous).
- Une boîte englobante est calculée autour de la plus grande région détectée.

3. Détection des points d'intérêt

- Dans la zone de la boîte englobante, des points caractéristiques sont détectés à l'aide de **detectMinEigenFeatures** (méthode Shi-Tomasi[16]).
- Ce sont des coins, stables et bien adaptés au suivi.

4. Initialisation du tracker

- Le tracker MATLAB **vision.PointTracker**(implémentation optimisée de Lucas-Kanade) est initialisé avec les points détectés.
- L'erreur maximale bidirectionnelle est fixée à 2 pixels pour garantir un suivi fiable.

5. Suivi dans la vidéo

- À chaque image, la fonction **step** met à jour la position des points.
- On visualise les points valides et on met à jour le tracker.
- Ce processus se poursuit jusqu'à la fin de la séquence vidéo.

3.2.4 Résultats attendu

Grâce à ce suivi visuel, plusieurs bénéfices sont obtenus :

- La position des zones d'analyse de la peinture reste stable et précise au fil du temps, même lorsque l'extincteur est en mouvement.
- Le système gagne en efficacité, car il n'est plus nécessaire de recalculer entièrement la détection de l'objet à chaque trame vidéo. Cela réduit la charge de traitement et améliore la réactivité en quasi temps réel.

La figure 8 illustre un exemple du suivi effectué à l'aide de l'algorithme de Lucas-Kanade, où les points caractéristiques de l'extincteur sont suivis d'une image à l'autre.

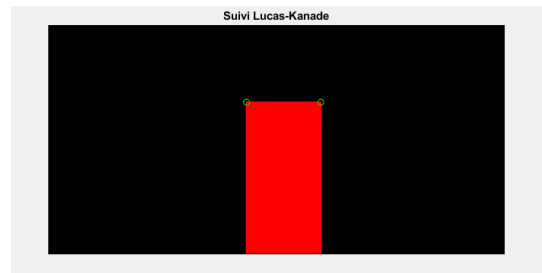


FIGURE 8 – Exemple de suivi des points caractéristiques de l'extincteur avec l'algorithme de Lucas-Kanade.

Les cercles verts montrent leur déplacement au cours du temps.

3.3 Extraction de la zone d'intérêt (ROI)

Dans le cadre d'un système de vision artificielle appliqué au contrôle qualité, il est primordial de délimiter avec précision la zone d'analyse dans chaque image. Cette zone, appelée région d'intérêt (Region of Interest ROI), correspond à la partie de l'image contenant l'information pertinente à évaluer, ici la surface peinte de l'extincteur.[9]

L'objectif principal de cette étape est donc d'isoler automatiquement la surface peinte sur laquelle sera appliquée l'analyse colorimétrique. En se focalisant uniquement sur la zone utile, on réduit considérablement le bruit visuel, on évite les erreurs d'interprétation, et on améliore la robustesse de l'évaluation de la teinte, notamment face aux perturbations liées à l'environnement industriel (fonds variés, ombres, reflets, présence d'objets métalliques ou de convoyeurs, etc.).

D'un point de vue technique, cette étape fait le lien entre la détection de la couleur rouge

(segmentation) et l'analyse colorimétrique proprement dite. Elle permet aussi de préparer le suivi temporel de l'objet en mouvement dans la séquence vidéo, en fournissant une localisation fiable et dynamique de l'extincteur à chaque image.[17]

En résumé, cette section répond aux objectifs suivants :

- Détecter et localiser précisément la zone de l'objet (l'extincteur) qui est effectivement peinte.
- Extraire cette zone automatiquement sous forme de ROI exploitable pour une mesure quantitative.
- Garantir une analyse ciblée sans interférence de l'environnement extérieur (fond, autres objets, reflets).
- Préparer les étapes suivantes (analyse de la teinte).

Dans ce qui suit, nous détaillons l'algorithme utilisé pour localiser cette zone, en s'appuyant sur les résultats de la segmentation, ainsi que les raisons qui motivent le choix de cette zone d'analyse.

3.3.1 Isolement de la partie peinte à analyser

Après avoir détecté la couleur rouge dans l'image, une image binaire est générée où les pixels correspondant à la couleur attendue (rouge vif) sont marqués en blanc. Cette image binaire (`red_mask`) contient donc la silhouette approximative de l'extincteur peint.

Pour extraire la **région d'intérêt (ROI)** contenant l'extincteur dans la vidéo, Pour extraire la zone exacte à analyser, nous utilisons la fonction (`'regionprops'`) de MATLAB, qui permet d'obtenir des informations géométriques sur les régions connectées de l'image binaire. L'attribut (`'BoundingBox'`) renvoie les coordonnées d'un rectangle minimal entourant la région blanche[9]. nous suivons une approche en plusieurs étapes :

1. Détection basée sur la couleur :

- Conversion de l'image en espace colorimétrique HSV (plus robuste aux variations lumineuses que RGB).
- Application d'un masque pour isoler les pixels rouges (typiques des extincteurs) :

$$\text{Masque} = (H < 0.05 \vee H > 0.95) \\ \wedge S > 0.4 \wedge V > 0.2$$

- Combinaison avec un masque RGB pour améliorer la détection .

2. Post-traitement du masque :

- Remplissage des trous (`imfill`) pour obtenir une région homogène.
- Fermeture morphologique (`imclose`) pour lisser les contours.
- Filtrage par taille (`bwareafilt`) pour ne conserver que la plus grande zone connexe.

3. Extraction du contour (Bounding Box) :

- Calcul de la boîte englobante (`regionprops`) autour de la région détectée.
- Découpage de l'image originale pour ne garder que la ROI.

La commande `bwareafilt(red_mask, 1)` est utilisée avant cela pour garder uniquement la plus grande région connectée, en supposant que l'extincteur est le plus grand objet rouge visible à l'écran.

Ensuite, l'instruction `imcrop(frame1, bbox)` permet de découper automatiquement la zone à analyser dans l'image originale. Cette région est ensuite utilisée pour effectuer des mesures colorimétriques précises.

Avantages de cette méthode

- Elle est automatique et adaptable : elle fonctionne même si la position de l'extincteur change dans la scène.
- Elle permet de réduire la complexité de l'image analysée, en éliminant l'arrière-plan et les détails inutiles.
- Elle rend les étapes suivantes (analyse de teinte, suivi) plus rapides et fiables.

3.3.2 Justification du choix de la zone

Le choix de cette boîte englobante centrée sur la zone rouge (représentant habituellement le corps cylindrique de l'extincteur) repose sur plusieurs justifications tant pratiques que techniques :

- **Proximité physique** : la surface centrale de l'extincteur est plus représentative de la qualité de la peinture que les parties métalliques (poignée, socle)
- **Évitement des perturbations** : étant donné que l'on ne tient pas compte de l'arrière-plan (convoyeur, ombres, lumière

ambiante), on évite une éventuelle erreur à cause des artefacts visuels.

- **Fiabilité de la mesure colorimétrique** : la mesure de la teinte (valeur moyenne de rouge dans la zone extraite) est beaucoup plus fiable appliquée à une zone homogène.
- **Suivi compatible** : cette localisation servira aussi à suivre l'objet au cours des images suivantes comme l'indique l'algorithme de Lucas-Kanade en définissant la région de détection initiale des points caractéristiques.

En somme, ce choix de zone va dans le sens de la précision, de la rapidité, et de la robustesse du système d'inspection automatique mis en place ici.[9]

Résultat attendu

Les figure 9 et figure 10 illustrent respectivement l'image d'entrée avant traitement et la zone d'intérêt (ROI) extraite, centrée sur la partie peinte de l'extincteur, utilisée pour l'analyse colorimétrique.



FIGURE 9 – Image avant l'extraction de la ROI.



FIGURE 10 – Image après l'extraction de la ROI

3.4 Analyse de la couleur : Calcul de la moyenne du rouge

Cette étape vise à quantifier l'intensité moyenne de la composante rouge dans la zone

d'intérêt (ROI), afin d'évaluer la qualité de la peinture appliquée sur la surface de l'extincteur.

3.4.1 Conversion RGB

Le modèle colorimétrique RGB (poursuivant une logique de synthèse additive) est sans doute l'un des systèmes de représentation des couleurs les plus usités dans le cadre des systèmes de traitement numérique d'images en nous appuyant sur le fait qu'un pixel est un triplet (R, G et B). Dans ce modèle, les valeurs R, G et B qui déterminent la couleur d'un pixel sont normalisées dans l'intervalle [0,1] (ou [0,255] selon la profondeur) et correspondent directement à :

R (Red) : composante rouge.

G (Green) : composante verte.

B (Blue) : composante bleue.

Ce modèle est principalement issu des dispositifs numériques d'acquisition d'images issus du numérique qui traduisent l'intensité lumineuse des couleurs via des filtres rouges, verts et bleus construites sur la logique de la synthèse additive de la couleur garantissant que le système RGB est le format brut correspondant à l'acquisition d'une image par une caméra.[9]. La figure 11 illustre la structure de l'espace colorimétrique RGB, où chaque couleur est représentée comme une combinaison des trois composantes de base : rouge, vert et bleu.

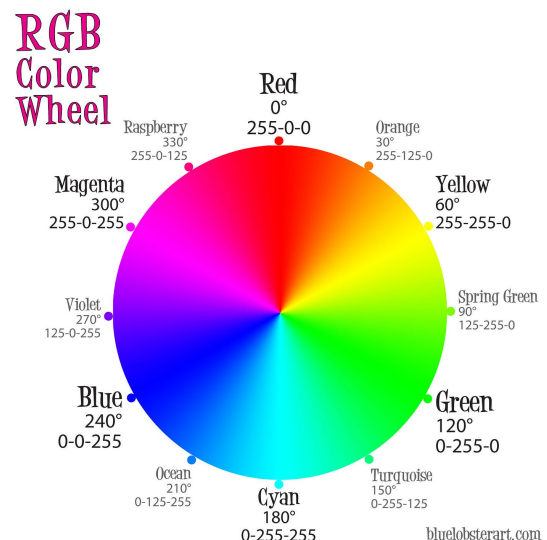


FIGURE 11 – Espace RGB

Le traitement d'images numériques en mode RGB permet de quantifier la luminosité d'une couleur, particulièrement utile dans le cadre du contrôle qualité sur l'intensité colorimétrique,

comme il est proposé dans notre projet. Cependant, la représentation RGB présente deux types de limites : d'une part le fait qu'elle ne soit pas perceptuellement uniforme (des distances non proportionnelles à des différences entre couleurs comme perçues par l'il humain). D'autre part, les composants YUV combinant des informations de teinte, de saturation et de nombre de luminance, quand on souhaite détecter une couleur donnée indépendamment de l'éclairement.

D'où l'usage courant d'un second espace de couleurs tel que l'HSV (Hue, Saturation, Value) pour isoler la teinte (la couleur perceptible indépendamment de l'éclairement) dont la détection se révèle plus robuste au détriment d'une analyse quantitative fine de l'intensité RGB.

Dans notre système de contrôle qualité de la peinture des extincteurs, nous avons adopté une **stratégie combinée** :

1. **Détection initiale** de l'objet peint à l'aide de masques de couleur dans l'espace HSV (plus tolérant aux variations d'éclairage).
2. **Extraction de la zone d'intérêt (ROI)** à partir du masque binaire détecté.
3. **Analyse colorimétrique détaillée** dans l'espace RGB, spécifiquement en étudiant la composante rouge.

Après avoir extrait la zone d'intérêt (ROI) correspondant à la surface visible de l'extincteur, nous extrayons la composante rouge (R) à l'aide de l'instruction suivante en MATLAB :

```
roi_red = roi(:, :,1)/255;
```

Cette opération nous permet de mesurer directement l'intensité de la peinture rouge dans la zone ciblée, de façon numérique, ce qui est fondamental pour une évaluation automatique de la qualité. En résumé :

- HSV est utilisé pour localiser la couleur rouge dans l'image,
- RGB est utilisé pour quantifier précisément l'intensité de cette couleur dans la région extraite.

Cette double approche optimise à la fois la robustesse de détection et la précision d'analyse, conformément aux bonnes pratiques en vision industrielle

3.4.2 Moyenne de la composante rouge dans la ROI

Au sein d'une image couleur RGB, chaque pixel est associé à un triplet [R,G,B] dans lequel R renseigne sur l'intensité de la couleur rouge. On appelle zone d'intérêt (ROI) une portion extraite de l'image, correspondant à une zone d'intérêt à analyser ici, la surface peinte de l'extincteur.

Le calcul de la moyenne de la composante rouge consiste au calcul de la moyenne de tous les pixels de la composante R uniquement, dans la ROI. Cela permet d'évaluer l'intensité moyenne de la peinture rouge appliquée sur la surface.[8] Mathématiquement, cela s'exprime comme :

$$\mu_R = \frac{1}{N} \sum_{i=1}^N R_i$$

où R_i est l'intensité rouge du $i^{\text{ème}}$ pixel dans la ROI, et N est le nombre total de pixels. Dans notre cas, la moyenne de la composante rouge est utilisée comme indicateur direct de la qualité de la peinture. Une peinture bien appliquée présente une couleur rouge uniforme et intense. Si la moyenne est faible, cela peut signifier :

- une **zone non peinte**.
- une **teinte insuffisante ou partielle**.
- un **reflet perturbant la détection**.

Dans notre algorithme MATLAB extrait la composante rouge et on calcule la moyenne :

```
mean_red = mean(roi_red(:));
```

Cette valeur est ensuite utilisée pour évaluer la qualité globale et aussi pour générer des histogrammes permettant une visualisation de la répartition des intensités rouges dans la zone analysée. La figure 12 représente une carte de chaleur des intensités rouges mesurées dans la zone d'intérêt, permettant de distinguer visuellement les régions bien peintes, mal peintes ou non colorées.

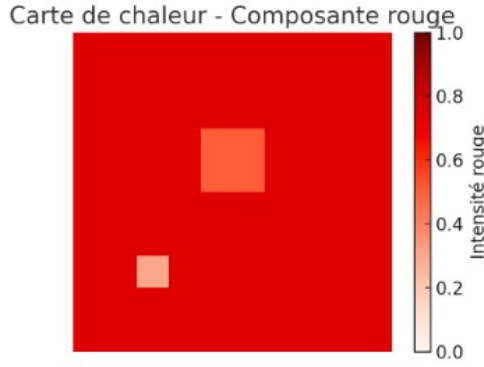


FIGURE 12 – Une carte de chaleur de l'intensité rouge dans la ROI (zones bien peintes, mal peintes et non colorées).

3.4.3 Seuils de décision pour juger la qualité

Des valeurs numériques que l'on appelle seuils de décision, permettent de classer ou de juger un phénomène selon un critère mesuré. En traitement d'image, des seuils de décision sont choisis pour revenir à la prise de décisions selon des propriétés extraites des pixels comme l'intensité lumineuse ou la couleur.

Ainsi pour l'évaluation de la qualité d'un objet coloré comme ici un extincteur peint, il est possible de définir des seuils sur l'intensité lumineuse moyenne de la couleur cible (rouge) que l'on souhaite obtenir ou des statistiques comme la variance, l'écart-type ou encore la proportion de pixels en dessous d'un certain niveau. Ces seuils de décision peuvent être choisis expérimentalement ou grâce à des méthodes d'optimisation afin d'être le plus fidèle possible à la perception visuelle humaine ou aux spécifications industrielles. [9]

Dans notre système de détection, les seuils ont été définis empiriquement après plusieurs essais sur des vidéos simulées. L'objectif était d'évaluer la qualité de la peinture rouge appliquée sur un extincteur qui se déplace horizontalement.

Nous avons calculé la **moyenne de la composante rouge** μ_R . Ensuite, nous avons identifié deux zones critiques dans l'image :

- **Les zones mal teintées** : pixels dont l'intensité de rouge est comprise entre un seuil de défaut ($S_d = 0.4$) et un seuil de qualité minimale acceptable ($S_q = 0.7$).
- **Les zones non colorées** : pixels dont l'intensité est inférieure à S_d .

À partir de ces seuils, l'évaluation de la qualité se fait comme suit :

- **Bonne qualité** : $\mu_R > S_q$ et moins de 5 % de pixels non colorés.
- **Qualité moyenne** : proportion de pixels mal teintés entre 5 % et 20 %.
- **Mauvaise qualité** : plus de 20 % de pixels non colorés.

Ces règles permettent une classification visuelle et statistique simple mais robuste, facilement interprétable par un opérateur ou un automate industriel. La figure 13 illustre l'histogramme des niveaux d'intensité de la composante rouge dans la zone analysée, avec des seuils empiriques indiqués pour distinguer les cas conformes (au-dessus de 0,7) et défectueux (en dessous de 0,4).

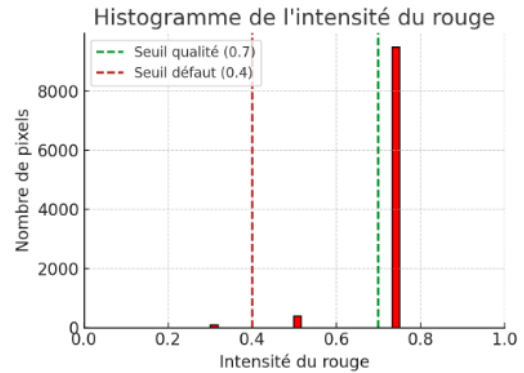


FIGURE 13 – un histogramme montrant la distribution des niveaux d'intensité du rouge, avec des lignes pour les seuils de qualité (0.7) et de défaut (0.4).

Justification des seuils

Les valeurs de 0,7 pour le seuil de qualité et 0,4 pour le seuil de défaut ont été choisies en fonction de la dynamique des intensités observées sur les extincteurs bien et mal peints. Ces seuils peuvent être ajustés selon l'environnement d'éclairage, la caméra utilisée ou les tolérances du client industriel.

Les figures figure 14, figure 15 et figure 16 montrent différentes interprétations visuelles de la qualité de la peinture à partir de l'intensité de la composante rouge dans la zone d'intérêt.

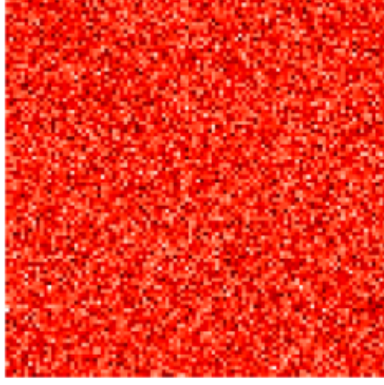


FIGURE 14 – Image représentant l'intensité de la composante rouge dans la zone d'intérêt.

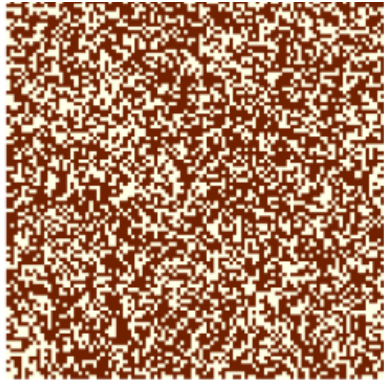


FIGURE 15 – Les zones mal teintées (intensité rouge insuffisante mais au-dessus du seuil de défaut).

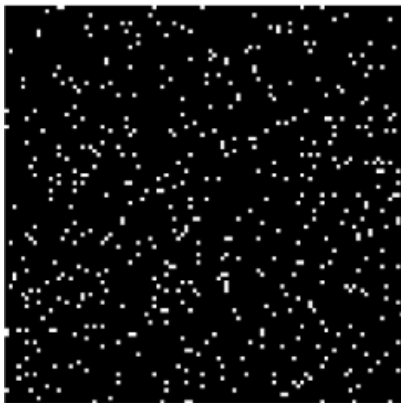


FIGURE 16 – Les zones non colorées (intensité rouge en-dessous du seuil critique).

Nous avons évalué ici la qualité de la peinture des extincteurs en mesurant sa couleur rouge, en analysant la composante rouge de l'image. Après une transformation d'image vers l'espace couleur RGB puis l'espace couleur HSV, nous avons isolé la région d'intérêt (ROI) correspondant à l'extincteur. La moyenne de la composante rouge de l'intensité de cette région a été comparée à des seuils

définis, permettant ainsi de qualifier la peinture comme étant de bonne qualité, moyenne, ou défectueuse : une méthode simple et rapide pour envisager un contrôle automatique de la qualité directement sur la vidéo.

4 Résultats expérimentaux

Cette section présente les résultats obtenus lors de l'évaluation du système proposé, en illustrant sa capacité à détecter automatiquement la qualité de la peinture appliquée sur les extincteurs.

4.1 Exemples de cas bons et défectueux

Afin de valider l'efficacité de l'algorithme développé, plusieurs vidéos ont été traitées. Les résultats montrent que le système est capable de distinguer avec précision les extincteurs bien peints des extincteurs présentant des défauts.

- Dans les cas bons, la moyenne de la composante rouge dépasse 0.7, et les zones non colorées sont inférieures à 5.
- Dans les cas défectueux, deux types d'anomalies sont détectées : les zones mal teintées (moyenne entre 0.4 et 0.7) et les zones non colorées (intensité < 0.4).

Afin d'illustrer ces cas, la Figure figure 17 montre un exemple d'extincteur bien peint, tandis que la Figure figure 18 présente un extincteur dont la peinture est de qualité moyenne ou défectueuse.

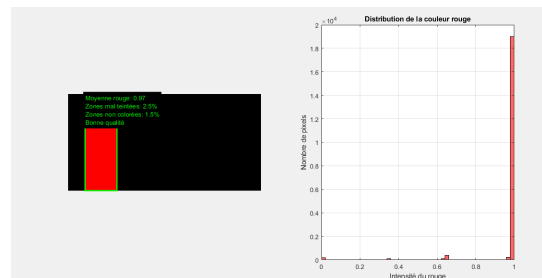


FIGURE 17 – Extincteur avec une qualité de peinture bonne (moyenne de la composante rouge $> 0,7$).

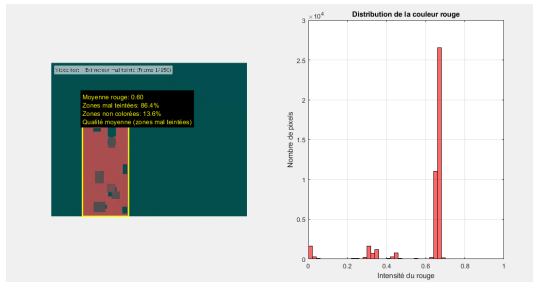


FIGURE 18 – Extincteur présentant une qualité moyenne ou défectueuse (moyenne de la composante rouge entre 0,4 et 0,7).

4.2 Images avant/après traitement

Le traitement d'image appliqué permet d'isoler efficacement la zone utile de l'image (l'extincteur), de filtrer les pixels non pertinents et d'analyser uniquement la surface peinte. Les figures suivantes montrent les images avant et après traitement, incluant la détection de la zone rouge, l'encadrement et l'histogramme.

Comme illustré dans figure 19, figure 20 et figure 21, la qualité de la peinture varie du très bon au mauvais.

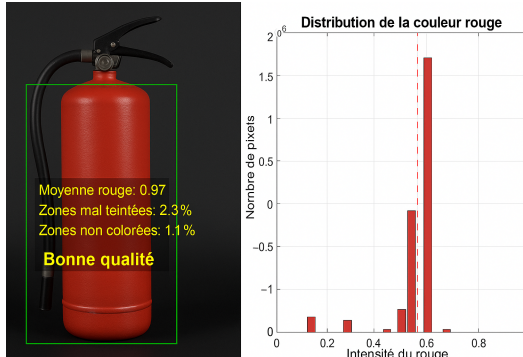


FIGURE 19 – Qualité très bonne de la peinture après traitement.

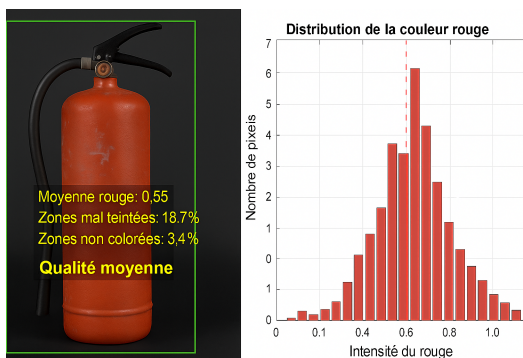


FIGURE 20 – Qualité moyenne de la peinture après traitement.

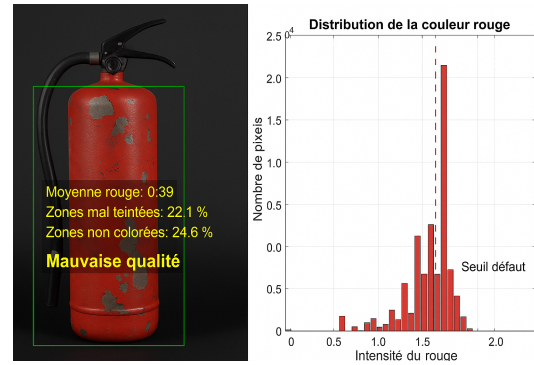


FIGURE 21 – Qualité mauvaise de la peinture après traitement.

4.3 Tableau de résultats

Afin de valider les performances de l'algorithme, plusieurs cas ont été analysés à partir des vidéos traitées. Chaque cas est évalué en fonction de la moyenne de la composante rouge, du pourcentage de zones mal teintées et non colorées.

Les résultats sont résumés dans tableau 3, où l'on observe une bonne corrélation entre les seuils définis et la décision finale prise par le système.

N°	moy Rouge	Mal T. (%)	Non C. (%)	Décision
1	0.97	2.3	1.1	Bonne
2	0.55	18.7	3.4	Moyenne
3	0.39	22.1	24.6	Mauvaise

TABLEAU 3 – Résultats d'évaluation de la qualité de peinture pour différents cas testés.

4.4 Taux de précision et efficacité

Le taux de précision et l'efficacité du système sont évalués en comparant les résultats de classification automatique avec des observations humaines, afin de mesurer la fiabilité et la robustesse de l'approche proposée. Sur l'ensemble des vidéos traitées, un total de 15 cas a été analysé :

- **12 cas bien classés** (qualité correctement évaluée par le système)
- **3 cas mal classés** (ex. : défauts mal identifiés ou marge d'erreur humaine)

Calcul du taux de précision :

$$\text{Taux de précision} = \frac{\text{Nombre de cas bien classés}}{\text{Nombre total de cas}}$$

$$\frac{12}{15} \times 100 = 80\%$$

5 Discussion

Cette section analyse et interprète les résultats expérimentaux obtenus, en mettant en évidence les points forts du système proposé, ses limites éventuelles, ainsi que les pistes d'amélioration envisageables pour une intégration industrielle optimale.

5.1 Avantages de la méthode

L'approche de détection automatique de la qualité de peinture présente plusieurs avantages significatifs par rapport à une inspection manuelle ou à d'autres techniques industrielles[18] :

Rapidité et automatisation : Grâce à l'analyse vidéo temps réel, la détection s'effectue automatiquement sans intervention humaine, ce qui augmente considérablement le débit de contrôle dans une chaîne de production.

Cohérence et objectivité : Contrairement à un inspecteur humain, le système basé sur des critères numériques (intensité rouge moyenne, seuils calibrés, etc.) est totalement neutre et répétable.

Coût réduit : L'utilisation d'une caméra Xbox 360, initialement conçue pour le jeu vidéo, représente une solution économique, tout en offrant une qualité suffisante pour cette application.

Facilité d'intégration : Le traitement en MATLAB permet une intégration rapide dans des environnements industriels existants, notamment avec des chaînes automatisées.

Analyse quantitative : Le système permet d'extraire des mesures précises (moyenne de la couleur, pourcentage de défauts, etc.), utiles pour l'amélioration continue de la qualité.

5.2 Comparaison avec une inspection humaine

Pour évaluer la pertinence et les avantages du système proposé, une comparaison a été réalisée entre l'inspection humaine traditionnelle et la méthode automatique mise en œuvre dans ce projet.

Comme présenté dans tableau 4, notre système automatique offre une vitesse de traitement bien supérieure, une objectivité constante et une fatigue inexistante par rapport à l'inspection humaine, bien qu'il puisse nécessiter des ajustements pour la détection de défauts subtils.

Critère	Inspection humaine	Système automatique (notre méthode)
Vitesse	Moyenne à lente	Élevée, en temps réel
Objectivité	Variable selon l'opérateur	Totale, basée sur des seuils fixes
Fatigabilité	Oui (erreurs après longue période)	Non (fonctionne 24/7 sans fatigue)
Capacité à détecter des défauts visuels subtils	Élevée (selon l'expérience)	Moyenne (nécessite des critères plus avancés pour des défauts fins)
Coût	Coût humain élevé (salaire, formation)	Faible coût matériel (caméra) et logiciel (MATLAB)

TABLEAU 4 – Comparaison entre l'inspection humaine et la méthode automatique

6 Conclusion

L'intégration d'un système automatique d'analyse de la qualité de peinture appliquée sur des extincteurs (dans un processus de peinture automatisé), a permis de concevoir et réaliser un système basé sur une caméra Kinect Xbox 360 et le logiciel MATLAB, dont la méthode est fondée sur la détection de la teinte rouge, sur le suivi de mouvement de type Lucas-Kanade, sur le zonage de la région d'intérêt (ROI) et sur l'analyse de la composante rouge dans le but d'apprécier la qualité de l'application de peinture.

Le système, à l'aide de seuils de décisions appropriés, va pouvoir opposer les extincteurs bien peints, mal peints ou non peints. Les résultats obtenus montrent de bon taux de réussite et une visibilité des défauts, réalisant les performances de l'inspection humaine avec une rapidité et une objectivité supérieure.

Ce travail montre qu'une telle technique peut s'intégrer à moindre coût dans un processus productif.

Références

- [1] Jean-Pierre Thomesse. *Les automatismes industriels*. Dunod, 5e édition édition, 2001.

- [2] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Cengage Learning, 4th edition, 2014.
- [3] MathWorks. *Computer Vision Toolbox Documentation*. MathWorks, May 2024. Consulté le 12 juin 2025.
- [4] N.E. Zain and W.N. Rahman. Three-dimensional (3d) scanning using microsoft® kinect® xbox 360® scanner for fabrication of 3d printed radiotherapy head phantom. <https://iopscience.iop.org/article/10.1088/1742-6596/1497/1/012005>, 2020. *Journal of Physics : Conference Series*, Volume 1497.
- [5] E. R. Davies. *Machine Vision : Theory, Algorithms, Practicalities*. Elsevier, 4th edition, 2012.
- [6] Kaifu Zhou, Taigang Liu, and Lifeng Zhou. Industry 4.0 : Towards future industrial opportunities and challenges. In *12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 2147–2152. IEEE, 2015.
- [7] Kouros Khoshelham. Microsoft kinect for xbox 360. https://www.researchgate.net/figure/Microsoft-Kinect-for-Xbox-360_fig1_281237173, 2011. Accessed : 12 June 2025.
- [8] Yang Liu, Yu Zhang, Jian Li, and Lei Wang. The impact of digital transformation on corporate environment performance : evidence from china. <https://www.nature.com/articles/s41598-024-63546-2>, 2024.
- [9] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson, 4 edition, 2018. ISBN : 978-0133356724.
- [10] David A. Forsyth and Jean Ponce. *Computer Vision : A Modern Approach*. Pearson, 2nd edition, 2012.
- [11] Pierre Soille. *Morphological Image Analysis*. Springer, Berlin, Heidelberg, 2003. ISBN : 978-3-540-42988-3.
- [12] Le Masyma. Tifo - morphologie mathématique (suite). https://lemasyma.github.io/cours/posts/tifo_morpho_maths_suite/, 2023. Consulté le 12 juin 2025.
- [13] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6) :679–698, 1986.
- [14] Bruce D Lucas and Takeo Kanade. *An iterative image registration technique with an application to stereo vision*. International Joint Conference on Artificial Intelligence, 1981. Proceedings of the 7th International Joint Conference on Artificial Intelligence.
- [15] Arnold WM Smeulders, Dung M Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah. Visual tracking : An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7) :1442–1468, 2014.
- [16] Jianbo Shi and Carlo Tomasi. *Good features to track*. IEEE, 1994. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [17] Xiang Li, Yiming Wang, Zhixin Chen, Jiawei Liu, and Wei Zhang. Enhancing the performance of perovskite solar cells through interface engineering : A review. <https://www.mdpi.com/2076-3417/15/10/5328>, 2025.
- [18] Richard Szeliski. *Computer Vision : Algorithms and Applications*. Springer, London, 2010. ISBN : 978-1-84882-934-3.