

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN MATHÉMATIQUES ET INFORMATIQUE
APPLIQUÉES

PAR
VINCENT LECOURS

DÉTECTION AUTOMATIQUE DES FACULTÉS AFFAIBLIES
SUR APPAREILS MOBILES

MAI 2015

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

Résumé

L'objectif de ce mémoire est l'analyse, la conception et la réalisation d'une application mobile permettant de déterminer si une personne a les facultés affaiblies suite à la consommation d'alcool, de drogue ou de toute autre substance pouvant changer son comportement normal.

Contrairement à plusieurs dispositifs déjà existants sur le marché aujourd'hui, cette application se base sur l'analyse vidéo du comportement des yeux afin d'établir un diagnostic. Basé sur des comportements innés de l'être humain, trois tests différents sont effectués afin de permettre d'émettre un diagnostic, suite à l'analyse des séquences vidéos prises par l'application du mouvement des yeux, de l'iris et la pupille de l'individu analysé. Les tests effectués sont ceux du nystagmus horizontal, la mesure de la dimension de la pupille suivant différents éclairage et la réaction et variation de la dimension de la pupille suite à un changement d'éclairage. Par la suite, les résultats de chacun des tests sont combinés afin de pouvoir émettre un diagnostic.

Afin d'expliquer et de détailler les différents algorithmes d'analyse d'image et du vidéo utilisés, plusieurs techniques d'analyse d'images sont abordées dont la binarisation, le calcul d'histogramme et la détection de caractéristiques dans une image suite à l'application de techniques de classificateurs automatiques.

Enfin, le développement de l'application mobile est abordé et a été réalisé sur la plateforme Windows pour différentes raisons expliquées dans le mémoire. Le processus de développement de l'application mobile, qui ne diffère que légèrement du processus d'application standard, est également traité en détail.

Table des matières

1	Introduction et présentation de la problématique de recherche	1
2	Revue de la littérature	6
2.1	Détection de facultés affaiblies	6
2.2	Analyse d'image	9
2.3	Développement mobile	19
3	Méthodologie et cadre expérimental	25
3.1	Choix de la plateforme	25
3.2	Présentation de la plateforme choisie	28
3.2.1	Survol de Windows RT et de son architecture	28
3.2.2	Sauvegarde des données	30
3.3	Algorithmes utilisés pour les différents tests	31
3.3.1	Algorithme général de détection de l'iris	31
3.3.2	Détails des étapes de l'algorithme	34
3.3.3	Mesure de la variation du diamètre de la pupille par rapport au temps (Réaction au changement de luminosité)	35
3.3.4	Mesure du diamètre de la pupille en condition de lumière stable (Repos)	35
3.3.5	Test du nystagmus horizontal	35
3.4	Présentation de l'application développée	36
3.5	Conclusion	41
4	Résultats et interprétation	42
4.1	Discussion	47
5	Conclusion	55
A	Procédure pour administrer le SFST de la NHTSA	68

<i>TABLE DES MATIÈRES</i>	iii
B Questionnaire actuel des symptômes de facultés affaiblies	70
C Spécifications techniques de la Surface 2 de Microsoft	72
Glossaire	74

Table des figures

2.1	(1) Représentation des 3 types d'ondelettes de Haar non-standard où (a) est verticale, (b) horizontale et (c) diagonale. (2) Représente la densité quadruple en 2D basé sur Haar. (Tiré de [73])	11
2.2	Représentation de deux caractéristiques sélectionnées par la technique du AdaBoost avec l'algorithme développé par Viola et Jones. (Tiré de [100])	13
2.3	Représentation de caractéristiques pseudo-Haar où 1(a), 1(b), 2(a), 2(b), 2(c), 2(d) et 3(a) sont énoncé par [73] et [100] tandis que les caractéristiques 1(c), 1(d), 2(e), 2(f), 2(g), 2(h) et 3(b) sont introduites dans [61] (Tiré de [61])	14
2.4	Dans l'ordre image originale d'un oeil (a), image de l'oeil ayant subi l'application du filtre de Canny (b)	15
2.5	Dans l'ordre image originale d'un oeil (a), image de l'oeil ayant subi une opération d'érosion (b), image de l'oeil ayant subi une opération de dilatation (c), image de l'oeil ayant subi une opération d'érosion suivi d'une dilatation (d), image de l'oeil ayant subi un traitement par un filtre de canny et une opération de dilatation par la suite (e).	18
2.6	Représentation schématique du fonctionnement d'une application régulière et de la même application dans un sandbox. (Tiré de [44])	22
3.1	Représentation vue du haut (a) et du côté (b) de la position d'un individu fixant la caméra en exécutant le test du Nystagmus.	27
3.2	Modèle général de traitement d'une MFT	30
3.3	Modèle général du Entity Framework	32
3.4	Logo de l'application Master	36
3.5	L'application Master installée sur Windows RT 8.1 à gauche et Windows Phone 8.1 à droite	37

3.6	Aperçu du projet de base dans Visual Studio 2013. On remarque deux projets distincts (<i>Master.Windows</i> et <i>Master.WindowsPhone</i>) et la présence d'un projet partagé parent qui regroupe le code et les objets communs aux deux projets enfants. On aperçoit également la présence du composant MFT intitulé <i>BACDetection</i>	38
3.7	Écran principal de l'application Master sur Windows RT 8.1 à gauche et Windows Phone 8.1 à droite	39
3.8	Écran du test du HGN de l'application Master sur Windows RT 8.1	40
3.9	Écran de l'application Master durant le test du HGN sur Windows RT 8.1	40
4.1	Graphique de la position angulaire de l'oeil par rapport à la caméra chez un individu non-intoxiqué. Les résultats obtenus (a) sont comparés avec eux obtenus dans [19] (b) pour un individu avec un pourcentage d'alcool de 0.00%. On peut remarquer la différence de nombres d'images captées pour le test ou (b) a un rythme de 10 images par secondes et (a) environ 4 causé par la charge de travail demandé simultanément à l'appareil par l'application et le système d'exploitation sur d'autres tâches diverses.	44
4.2	Graphique de la position de l'oeil par rapport à la caméra chez un individu commençant à présenter des symptômes de facultés affaiblies avec un taux d'alcool d'environ 0.08%. Les résultats obtenus (a) sont comparés avec eux obtenus dans [19] (b) pour un individu avec un pourcentage d'alcool de 0.09%. On peut observer la même tendance dans les deux expériences malgré des taux de captures différents, mais une durée avoisinant les 8 secondes dans les 2 cas.	45
4.3	Graphique de la position de l'oeil par rapport à la caméra chez un individu présentant des symptômes de facultés affaiblies avec un taux d'alcool d'environ 0.12%. Les résultats obtenus (a) sont comparés avec ceux obtenus dans [19] (b) pour un individu avec un pourcentage d'alcool de 0.22%. Bien qu'il y ait une différence importante entre le pourcentage d'alcool présenté ici et celui par [19], il est possible de voir que la tendance est sensiblement la même, malgré des taux de capture différents, mais une durée sensiblement identique (8 secondes).	46
4.4	Exemples de 2 images qui illustrent les états de départ dans l'obscurité (a) et de fin dans la lumière (b) du test de dilatation/contraction de la pupille.	48

4.5	<i>Exemple d'un graphique de la variation du diamètre de la pupille par rapport au temps lors d'un changement brusque d'éclairage chez un individu non-intoxiqué. Puisque l'on capture à environ 12 images/seconde et que la réaction s'effectue sur 12 images, le test est passé avec succès par l'individu. Dans le cas présenté ici, le changement de lumière commence à la 30e image du vidéo et se termine à la 42e.</i>	49
4.6	<i>Image d'une séquence où un utilisateur passe le test du HGN</i>	50
4.7	<i>Image d'une séquence où le test se passe dans le noir pour vérifier le temps de réaction de la pupille. On aperçoit également la difficulté à différencier l'iris de couleur brun à la pupille.</i>	51
4.8	<i>Représentation des prises de vues à effectuer à l'aide d'un damier lors de la calibration d'une caméra.</i>	52
4.9	<i>Deux captures d'écran de l'application illustrant la nécessité d'avoir un environnement sombre, mais une source de lumière d'ambiance devant l'individu et derrière l'appareil.</i>	54

Liste d'algorithmes

1	Algorithme de détection de la pupille	33
---	---	----

Remerciements

Je tiens à remercier mon directeur de recherche, François Meunier, professeur au département de Mathématiques de d'Informatique (DMI) de l'UQTR, pour sa patience, sa persévérance et ses conseils tout au long de mon programme de maîtrise.

J'aimerais également remercier mon entourage, plus particulièrement ma mère, Sylvie Noël, pour ses encouragements tout au long du processus de mes études universitaires.

Finalement, j'aimerais remercier les professeurs Alain Chalifour, Philippe Boissonnet (UQTR) et Éric Busvelle (Université de Bourgogne) de m'avoir introduit à la recherche scientifique dans le cadre de stages effectués à l'Université de Bourgogne lors de mes études au Baccalauréat.

Chapitre 1

Introduction et présentation de la problématique de recherche

Cette recherche s'intéresse à la détection des facultés affaiblies chez une personne au moyen d'un appareil mobile. Il est connu depuis plusieurs siècles que l'ingestion de différentes substances, comme l'alcool, peut modifier le comportement normal humain. Certaines substances peuvent affecter un seul sens comme la vue ou l'ouïe, tandis que d'autres peuvent affecter plusieurs sens à la fois. La recherche présentée ici se contente d'analyser les effets de l'alcool et de certaines drogues sur le comportement oculaire humain, bien que les mêmes symptômes peuvent être retrouvés avec l'ingestion d'autres substances.

Les débuts de la production et la consommation d'alcool remontent à l'époque du néolithique lorsque l'Homme devient sédentaire et commence à pratiquer l'agriculture. Par la suite, les boissons alcoolisées ont joué plusieurs rôles dans les différentes civilisations de l'Antiquité. Les Égyptiens et les Romains s'en servaient pour les fêtes et les divertissements ainsi que pour ses propriétés antiseptiques, les Grecs s'en servaient pour leurs assemblées publiques qui servaient à discuter du fonctionnement de la société et les Babyloniens fabriquaient plus de 16 variétés de bières. Malgré tout, dans toutes ces sociétés, l'ivresse n'est pas bien vue aux yeux de la population et des hommes tels que Jules César sont bien vus compte tenu de leur consommation modérée.

Comparativement à la consommation d'alcool qui est présente dans les dif-

férentes civilisations présentes il y a plusieurs millénaires, la notion d'alcoolisme et de facultés affaiblies sont relativement récentes. Un médecin suédois du nom de Magnus Huss qui décrira en 1849 « les manifestations pathologiques de l'intoxication alcoolique » après avoir découvert que plusieurs maladies, dont la population hospitalière de Stockholm était victime, étaient directement lié à une surconsommation d'alcool.

Dans les années 1920, afin de diminuer les maladies liés à l'alcool et de réduire les taux de criminalité dans les quartiers pauvres, le gouvernement américain lance la prohibition qui interdit la production et la distribution de l'alcool. Plus près de chez nous, le gouvernement québécois tiendra un référendum sur la prohibition de l'alcool en 1919, où le "Oui" l'emportera confortablement. Par contre, face à la grogne populaire, le gouvernement reviendra sur sa décision quelques mois plus tard et abolira finalement la prohibition en 1921 et créé la Commission des Liqueurs du Québec (maintenant devenu la SAQ (Société des Alcools du Québec)). (Jacques Lacoursière [53]) Le gouvernement américain reviendra lui aussi sur sa décision et cessera d'appliquer la prohibition en 1933.

L'après-guerre amène bien des changements dans la société nord-américaine en général, y compris dans la société québécoise. Le nombre croissant de véhicules automobiles amène la construction de nouvelles infrastructures et de nouvelles routes. Avec ces changements et étant donné que la consommation de bière et de spiritueux est bien ancré dans les moeurs de la société, on voit l'avènement d'un nouveau problème : l'ivresse au volant. Bien que le code de la sécurité routière interdit la combinaison de la consommation d'alcool et de la conduite d'un véhicule motorisé, peu de policiers appliquent la loi jusqu'en 1969 avec l'adoption, par le gouvernement canadien, du "bill Omnibus" qui fixe le seuil de tolérance à 80 mg/100 mL d'alcool dans le sang. (Parlement Canadien [70]) Au Québec, depuis 1970, on utilise un Alcootest qui analyse l'haleine de l'individu visé afin de détecter la présence d'un taux d'alcool supérieur à la limite permise. Bien que souvent contesté, il s'agit d'un outil fiable, mais pas infaillible.

Selon certaines statistiques de la SAAQ (Société de l'Assurance Automobile du Québec), grâce aux multiples campagnes de préventions au fil des ans, le bilan routier dans les accidents où l'alcool est en cause s'est amélioré grandement. En 1987, la proportion des décès dans des accidents qui impliquaient l'alcool était de 50% tandis qu'elle est de 37% en 2011. (Radio-Canada [78]) Par contre, les jeunes de 20 à 24 ans sont toujours ceux qui

présentent le plus haut nombre d'infractions pour la conduite avec les facultés affaiblies. (Samuel Perreault [74]) Ce n'est donc pas une surprise que la plupart des campagnes de publicités de la SAAQ pour l'alcool au volant visent directement ce groupe d'âge.

Avec l'avènement de nouvelles technologies, les moyens de sensibiliser la population évoluent aussi. Auparavant, les journaux, la télévision et la radio étaient les moyens à la mode afin de passer des messages. En 2011, la SAAQ publie Code-Bar, une application pour appareils mobiles pouvant également se brancher sur Facebook et permettant aux jeunes de contrôler leur consommation d'alcool et de se trouver un conducteur désigné dans leur entourage. (SAAQ [22]) Par contre, la SAAQ n'est pas la seule à publier ce genre d'applications. Que ce soit par la publication d'applications rigolotes afin de calculer son nombre de consommations ou que ce soit avec la sortie d'applications officielles permettant de faire de la sensibilisation auprès des utilisateurs, plusieurs personnes tentent de faire passer le message que la modération dans la consommation d'alcool est souvent préférable.

Avec l'avènement massif du téléphone intelligent notamment les Blackberry au début des années 2000, mais surtout avec la sortie du iPhone original en 2007, ces appareils ont changé totalement le paysage qu'Internet avait lors de leur sortie. Les courriels, messages textes, GPS, les réseaux sociaux sont désormais choses communes. Les technologies du Web se sont également adaptées. Des sites interactifs pouvant être vus sur un petit écran de 3.5" sont maintenant chose courante et surtout sont développés avec des technologies respectant les standards du W3C (World Wide Web Consortium) et ne nécessitant pas l'installation de plugiciels externes.

Suite à la création de nouvelles catégories d'appareils caractérisés par leurs fonctionnalités et leur mobilité, de toutes nouvelles façons de créer et de distribuer des logiciels sont apparues. Auparavant, la création de logiciels était surtout réservée aux grandes entreprises ayant les moyens d'avoir des équipes dédiés aux ventes et au marketing tandis que maintenant les outils de développement sont ouverts à tous et si une personne a une idée de logiciel, il peut la développer et la publier dans un magasin d'applications ou les utilisateurs pourront ensuite télécharger l'application créée par un individu ayant eu l'idée pendant que ce dernier reçoit un certain pourcentage des ventes. Cette façon de faire a changé complètement le monde du développement de logiciels et permis à plusieurs personnes et à plusieurs nouvelles philosophies de soit apparaître ou soit de mûrir comme c'est le cas avec

la stratégie de développement Agile.

Contrairement à l'apparition de nouvelles technologies qui ne changent que la façon dont les gens en technologies de l'information (TI) travaillent, l'introduction du téléphone et de la tablette intelligente a permis de démocratiser l'accès à Internet pour plusieurs personnes et a également permis à plusieurs autres de toujours rester connectées à leur entourage et à leur travail. Puisque l'appareil est mobile et peut suivre l'utilisateur en tout temps, de nombreuses applications ont été créées qui n'auraient aucune raison d'être sur un ordinateur personnel standard et vice-versa. En effet, selon une étude publiée par la firme ComScore en octobre 2014 [66], 49% du temps que les Canadiens passent en ligne l'est sur un appareil mobile (téléphone ou tablette) et maintenant environ 4% n'utilisent qu'un appareil mobile sans être en ligne. De plus, profitant de l'avancée de certaines technologies matérielles comme une caméra photo et vidéo et de certains capteurs, tels que le gyroscope, GPS, baromètre, etc.) et d'un prix relativement peu élevé, le rythme de dévoilement de ces nouveaux appareils est souvent bi-annuel ou annuel, ce qui amène plusieurs personnes à utiliser les toutes dernières technologies disponibles sur le marché.

Démarrant originalement avec des spécifications assez faibles et profitant de l'avancée rapide des technologies en miniaturisation des composants électroniques, il est assez facile de dire que les spécifications de certains appareils rivalisent avec celles des ordinateurs modernes tout en partageant le même genre de technologies. Processeurs multi-coeurs, caméras haute-résolution, écrans haute-définition, etc. Ceci implique donc que certaines personnes commencent à considérer prioritaire l'achat d'un appareil mobile versus l'achat d'un ordinateur de bureau ou d'un ordinateur portable.

Bien que les domaines de la biologie humaine, l'informatique et la justice ne semblent pas avoir de points communs facilement visibles, il est tout de même possible de faire certains rapprochements. Grâce aux études effectuées sur le comportement humain lors de la consommation d'alcool, il semble possible de développer un système informatique permettant d'analyser certaines composantes de ce comportement, telles que les réflexes oculaires, et d'apporter un diagnostic. Selon Jessica LeFevre[60], bien que le test du Nystagmus soit efficace dans bien des situations, il est souvent contesté en Cour puisque ce test est administré par un policier et il n'y a pas de méthode scientifique et un milieu stable afin de faire ce test. Même son de cloche du côté de Frederick Leatherman [59], qui lui, ajoute le fait que le témoin doit

avoir reçu une formation pour être en mesure d'utiliser l'appareil servant à évaluer son taux d'alcoolémie dans le sang. L'introduction d'une méthode stricte et rigoureuse ainsi qu'un test normalisé et administré par une machine dans un environnement contrôlé aurait comme conséquence d'augmenter la crédibilité du test devant un juge.

Ce mémoire tente de répondre à trois questions principales :

- Est-il possible de détecter, avec précision et efficacité, la présence de facultés affaiblies chez l'Homme à partir de séquences vidéo de l'oeil ?
- Est-il possible d'effectuer, efficacement, de l'analyse vidéo à partir d'un appareil mobile moderne ?
- Est-il possible d'effectuer l'analyse du mouvement des yeux à partir d'une caméra d'un appareil mobile moderne afin de détecter la présence de facultés affaiblies ?

Donc, afin de répondre à la problématique de recherche énoncée, cette étude comporte 3 chapitres. Le premier met en évidence une revue critique de la littérature relative aux deux sujets principaux abordés tout au long de cette recherche soit la détection des facultés affaiblies chez l'Homme ainsi que les différentes techniques d'analyse d'image utilisées pour d'abord localiser les pupilles dans les séquences vidéo et ensuite déduire des mesures caractérisant le comportement des pupilles, comme par exemple le diamètre et l'angle de visée. Le second présente la méthodologie ainsi que le cadre expérimental utilisé pour cette recherche ainsi que plusieurs détails techniques sur l'application développée afin de répondre à la problématique de départ. La troisième partie présente les résultats obtenus ainsi qu'une interprétation des différents résultats ainsi qu'une discussion sur les apports et les limites de cette recherche. En guise de conclusion, une synthèse des faits saillants de la recherche et développements futurs sont présentés.

Chapitre 2

Revue de la littérature

Cette partie passera en revue les différents concepts qui seront utilisés et étudiés au cours de cette recherche. Premièrement, le sujet de la détection de facultés affaiblies sera abordé et ensuite le chapitre se concentrera sur les différentes techniques d'analyse d'image existantes et utilisées dans le cadre de ce travail, pour finalement se terminer sur une revue des différents défis, problématiques et différences du développement d'applications mobiles modernes versus le développement d'applications traditionnelles pour ordinateur de bureau.

2.1 Détection de facultés affaiblies

Plusieurs sources définissent l'état de facultés affaiblies chez un individu de façons différentes. Pour la plupart des Québécois, la notion de facultés affaiblies est directement liée avec la notion de quantité limite d'alcool dans le sang légale pour être en mesure de conduire un véhicule motorisé soit 8 mg/100 mL ou très souvent identifiée par « .08 » (0.08 g/L). Pour le Thésaurus de l'activité gouvernementale du Gouvernement du Québec [28], il s'agit d'un état où « ses capacités sont diminuées en raison d'une consommation d'alcool ou de drogue ». Pour la SAAQ [21], en plus de la consommation d'alcool et de drogues, on ajoute la consommation de médicaments qu'ils soient sous ordonnance ou non. Même chose du côté de la Gendarmerie Royale du Canada (GRC) [26].

Bien que souvent liée seulement à l'alcool dans la conception culturelle québécoise et canadienne, la notion de facultés affaiblies s'applique aussi aux drogues et aux médicaments pour les 3 organismes. Par contre, ce travail

ne prends en compte que la détection de l'alcool, malgré que souvent, les réactions aux drogues et médicaments sont souvent semblables.

Depuis quelques années, la SAAQ fait de la sensibilisation auprès des conducteurs québécois afin de tenter de prévenir des morts reliées à la fatigue au volant [96]. Bien que les symptômes de la fatigue au volant s'apparentent beaucoup à ceux éprouvés lorsqu'un individu est sous l'effet d'une substance, on ne considère pas la fatigue au volant comme étant des facultés affaiblies. Dans ses publications, la SAAQ mentionne « Entre 17 et 19 heures d'éveil, vos capacités physiques et mentales équivalent à celles d'une personne ayant un taux d'alcool dans le sang de 50 mg / 100 ml (0,05) » [23]. Bien que ce soit le cas, selon Citek et al. [14], les individus déprivés de sommeil passent avec succès les tests « Field Sobriety Test » (FST) utilisés par les policiers afin de détecter une personne souffrant des facultés affaiblies. Donc, bien que les symptômes soient similaires, impossible d'inclure la fatigue ou sommeil au volant comme faisant partie des sources d'un individu étant sous l'effet des facultés affaiblies.

Mentionné précédemment, les FST sont une batterie de tests développés par la NHTSA (National Highway Traffic Safety Administration). Les premiers de ces tests sont apparus dans l'étude commandé par l'organisme fédéral en 1975 et publiés dans l'étude de Harris et al. [39], en 1980. Au total, cette étude présentera 20 tests ou indices étant observables chez un individu lorsqu'il dépasse la limite de .10. Parmi ces tests, on retrouve entre autre le test du nystagmus horizontal, le toucher du nez avec un doigt, le traçage d'un parcours de labyrinthe, le comptage de nombre en ordre décroissant.

Afin d'établir une norme standardisée par rapport aux tests auxquels les conducteurs soupçonnés d'être en état d'ébriété doivent se soumettre, la NHTSA a choisi, durant les années 80, les trois tests les plus significatifs, et corrélés entre eux, dans la détection des facultés affaiblies chez un individu soit le test du Nystagmus horizontal (HGN), le test d'équilibre sur une jambe (One-Leg-Stand) et le test de marcher et tourner (Walk and turn). La procédure à employer par les policiers afin de faire passer ces trois tests est présentée à l'annexe 1 [3].

Bien que ces trois tests font partie de la batterie de tests SFST (Standardized Field Sobriety Tests), un seul est basé sur des faits scientifiques, soit celui du nystagmus horizontal (HGN) Selon la NHTSA [46], le nystagmus horizontal se définit comme étant un mouvement involontaire de l'oeil qui

ressemble à un rebondissement de l'oeil étant causé par le fait que l'oeil tente de se réajuster après un déplacement. Ce phénomène se produit généralement lorsqu'un individu est intoxiqué par une substance et que l'oeil tente de regarder à un angle de 45 degrés. Cette relation entre l'alcool et le phénomène du Nystagmus a été énoncé en premier par Aschan [5] et ensuite par Wilkinson et al. [105] ont découvert la relation proportionnelle entre la consommation d'alcool et la présence du nystagmus chez un individu.

Bien que rarement mentionné, le nystagmus se manifeste également d'une façon verticale tel que mentionné dans [60] et [65].

Malgré qu'approuvé par la NHTSA, le test du nystagmus a continuellement été attaqué en cour pour diverses raisons. Bien que tous s'entendent pour dire que l'alcool joue un rôle dans la présence du nystagmus, plusieurs autres facteurs peuvent également entrer en ligne de compte et faire apparaître ce phénomène oculaire. Plusieurs auteurs autant au niveau judiciaire qu'au niveau scientifique contestent les conclusions et l'application de ce test, comme c'est le cas dans LeFevre [60], Booker [8] et Rubenzer [83]. Un des arguments, les plus souvent amenés par les détracteurs de ce test, est que les conditions lorsque le test est administré ne sont pas celles prescrites par la NHTSA et donc le test ne peut être valide. On attaque même, souvent, la compétence du policier qui a administré le test, l'accusant d'avoir déjà un parti pris avant même le début du test et donc d'influencer son résultat.

Un autre élément également amené est le fait que la NHTSA tenterait de camoufler certaines études qui démontrerait à tort l'apparition du nystagmus lors de la consommation excessive d'alcool. Il est par contre vrai qu'un individu peut avoir une présence de nystagmus sans la consommation d'alcool. Il peut être causé également par la prise de certains médicaments ou même certains individus présentent cette condition depuis leur naissance. Malgré ceci, certains juges lient automatiquement l'apparition de nystagmus au fait de la consommation d'alcool et donc jugent l'individu coupable sans qu'il puisse présenter une défense juste.

D'autres indices permettent également de détecter l'ingestion de substance dans le corps par l'analyse des iris et des pupilles comme mentionné dans [91] et [65]. Les tests nécessitent d'analyser la réaction de la pupille face à un changement brusque de luminosité. Dans un état normal, le diamètre de la pupille devient grand devant un environnement sombre et devient petite dans un environnement éclairé. Par contre, sous l'effet de drogues ou d'al-

cool, ce comportement change. La pupille réagit beaucoup moins rapidement à un changement brusque de lumière et peut même, dans certains cas, ne pas réagir du tout.

Puisqu'il s'agit du test le plus susceptible d'être analysable via un appareil mobile et celui étant le plus fiable, le test utilisé dans l'application mobile développé dans le cadre de ce travail sera celui du HGN accompagné des tests de mesure de la dimension de la pupille ainsi que de la variation du diamètre de la pupille à la réaction à la lumière.

2.2 Analyse d'image

L'analyse d'image est une discipline qui unit les mathématiques et l'informatique, puisque le traitement d'image est souvent basé sur des équations mathématiques étant donné qu'une image est représentée informatiquement par une ou plusieurs matrices. Cette section présente des techniques et des approches d'analyse qui sont considérées dans le développement de l'application tout en se spécialisant dans l'analyse d'image de visages, d'iris et de pupilles.

Une technique simple, mais souvent efficace, de recherche d'images est la recherche par histogrammes de couleur. Bien que l'on définit les histogrammes de couleur RGB comme étant des espace en trois dimensions, la théorie s'applique souvent aux images en niveaux de gris à l'aide des valeurs d'intensité de couleur qui forment un histogramme à une dimension que l'on définit comme histogramme d'intensité. Comme décrite dans Chakravarti et al. [12] et Hussain et al. [13], la technique est relativement simple. La première étape consiste à bâtir une bibliothèque d'images avec des résultats positifs et en stockant leur résultats d'histogramme. Par la suite, la recherche d'image se fait en comparant l'histogramme de l'image recherché aux histogrammes présents dans la bibliothèque en calculant la distance euclidienne entre ces deux histogrammes. Bien que cette technique soit conceptuellement simple, plusieurs chercheurs ont amené des précisions et développé la technique. Un des premiers a été Otsu dans un article publié en 1979 [72] qui a défini une technique de sélection basé sur un histogramme de niveau de gris permettant la détermination du seuil optimal séparant les deux modes d'un histogramme bimodal.

Certains chercheurs se sont penchés sur l'amélioration de la technique de

l'histogramme comme Laptev, dans 2 articles publiés en 2006 [56] et 2009 [57], qui a développé deux nouvelles techniques, soit l'utilisation d'histogrammes locaux afin d'identifier des zones d'intérêts ainsi que l'utilisation de l'apprentissage automatique via l'algorithme AdaBoost.

Developpée par Freund et al. et publiée en 1997 [36], la technique nommée "Adaptative Boosting" ou AdaBoost est une technique d'apprentissage automatique, basé sur la théorie du Boosting, qui repose sur la sélection itérative de classificateurs faibles en fonction de la distribution de l'échantillon d'apprentissage. Cette technique est représentée mathématiquement par le classificateur binaire H tel que

$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right) \quad (2.1)$$

où T est une combinaison de classificateurs faibles h_t combiné à un poids α_t . Chaque classificateur faible est défini par

$$h(z) = \begin{cases} 1 & \text{si } g(f(z)) > \theta \\ -1 & \text{sinon} \end{cases} \quad (2.2)$$

Dans le contexte d'analyse d'image et de localisation d'objets, on définit f en terme des propriétés de l'image sur des régions r et on utilise la technique d'AdaBoost pour sélectionner des particularités afin de maximiser la performance de classification de l'échantillon de la région r . Cette dernière technique a été un des apports de Viola et al. dans leur article de 2001 sur la détection de visage [100] présenté plus loin dans ce texte.

En 1998, Papageorgiou et al. publient un article [73] définissant plusieurs nouvelles notions dans la détection d'objets dans une image. L'apport majeur de cet article est l'introduction des notions des caractéristiques pseudo-Haar telles que représentées à la figure 2.1. Dans cette figure, (1) présente les 3 types d'ondelettes en deux dimensions pseudo-Haar soit verticale (a), horizontale (b) et diagonale (c). Dans la deuxième partie de cette figure (2), les ondelettes en deux dimensions et à densité quadruple sont présentées. Cet apport permet de mieux représenter et identifier les caractéristiques à retenir des images analysées afin de pouvoir détecter la présence de caractéristiques précises ou même d'objets.

Bien que nommées d'après Alfred Haar, ces caractéristiques portent ce nom

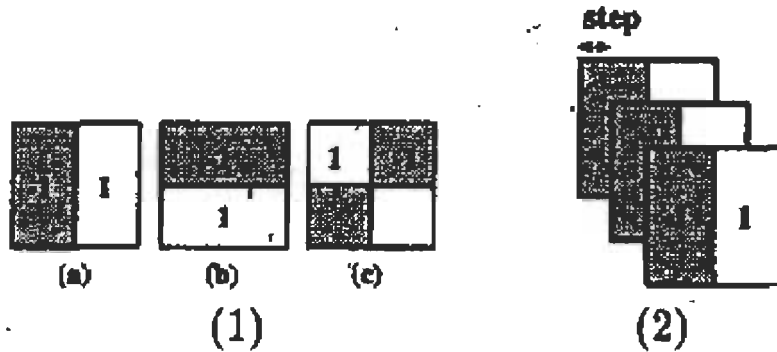


FIGURE 2.1 – (1) Représentation des 3 types d'ondelettes de Haar non-standard où (a) est verticale, (b) horizontale et (c) diagonale. (2) Représente la densité quadruple en 2D basé sur Haar. (Tiré de [73])

à cause de leur ressemblance avec les ondelettes de Haar développées par ce mathématicien. La technique consiste à trouver et à mémoriser des caractéristiques basées sur les ondelettes de Haar afin de pouvoir les retrouver sur une image à analyser.

S'appuyant sur les travaux effectués par Papageorgiou et al., Paul Viola et Michael Jones ont publié, en 2001, un article [100] qui change la manière dont la détection d'objets, mais surtout de visage, est effectuée et vue dans le milieu scientifique. En plus de présenter une technique de détection nouvelle, elle apportait également une façon de faire cette détection en temps réel sur un ordinateur muni d'un processeur Pentium III de 700 Mhz qui était la norme du temps. Afin d'y arriver, leur article présente en détail les trois nouveautés utilisées dans cette détection soit : l'utilisation de la notion d'image intégrale, l'utilisation de l'algorithme d'apprentissage AdaBoost en détection d'objets ainsi que l'utilisation de classificateurs en cascade.

La notion d'image intégrale utilisée par Viola et Jones découle des recherches de Franklin C. Crow publiées dans un article de 1984 [18]. Initialement, la notion d'image intégrale était utilisée afin d'optimiser l'affichage et le calcul de texture à appliquer dans le graphisme par ordinateur. Viola et Jones ont adapté cette notion à la détection d'objets. En plus de permettre de localiser facilement les régions d'intérêt dans l'image analysée grâce à la sommation des points d'intérêts dans des rectangles de plus petite taille, le calcul de

l'image intégrale est un algorithme de type $O(n)$, qui se prête alors très bien à l'exécution en temps réel.

Soit une image i , on calcule son image intégrale I du point (x,y) en effectuant les opérations suivantes :

$$I(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2.3)$$

et donc

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2.4)$$

$$I(x, y) = I(x - 1, y) + s(x, y) \quad (2.5)$$

En combinant ces deux dernières équations, on obtient :

$$I(x, y) = i(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1) \quad (2.6)$$

L'utilisation de l'algorithme d'apprentissage AdaBoost dans le contexte d'analyse d'image est une idée reprise des travaux précédents de Tieu et Viola [98]. Dans le contexte de cet article, le classificateur faible $h_j(x)$ est défini comme

$$h(x) = \begin{cases} 1 & \text{si } p_j f_j(x) < p_j \theta_j \\ 0 & \text{sinon} \end{cases} \quad (2.7)$$

où x est une région rectangulaire de 24×24 pixels sur laquelle est effectuée l'apprentissage, f_j est une caractéristique et p_j est l'élément de parité qui indique la direction de l'inégalité du cas de valeur 1. Un exemple de résultats appliqués à la détection de visage est présenté à la figure 2.2.

Afin de pouvoir détecter les visages et autres objets, il est nécessaire de comparer les caractéristiques apprises à l'aide de l'apprentissage effectuée à partir de l'échantillon, à notre image traitée. Puisque l'objet à détecter n'est pas nécessairement à la même distance de la caméra que les images de l'échantillon, il est nécessaire de faire la détection à différentes échelles. Tout ce travail est donc très intensif en calcul à effectuer par la machine. Ceci est diminué par la troisième avancée de Viola et Jones soit la cascade de classificateurs. Le principe de cette cascade est fort simple. Premièrement, on fait la supposition que la plupart des caractéristiques seront fausses lorsque analysées sur l'image, donc on commence par analyser les classificateurs les plus simples, soit ceux qui ont besoin du moins de calculs. Aussitôt que la

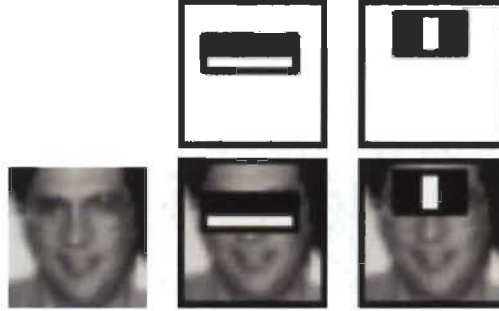


FIGURE 2.2 – Représentation de deux caractéristiques sélectionnées par la technique du AdaBoost avec l’algorithme développé par Viola et Jones. (Tiré de [100])

caractéristique n’est pas détecté, elle est rejetée. On peut donc qualifier cette cascade comme étant un arbre de décision dégénéré, soit un arbre où chaque noeud ne contient qu’une seule branche.

En 2002, Lienhart et Maydt publient un article [61] venant compléter les travaux de Viola et Jones en affirmant diminuer les fausses détections d’objets en effectuant une rotation de 45 degrés sur les caractéristiques pseudo-Haar, tel que montré à la figure 2.3. Dans leur article, ils apportent également une modification au calcul de l’image intégrale afin de l’adapter à leur technique. L’image intégrale d’un point (x,y) est définie dans [61] comme étant :

$$I(x, y) = \sum_{x' \leq x, x' \leq x - |y - y'|} i(x', y') \quad (2.8)$$

et donc la somme pour un rectangle est :

$$Somme(r) = I(x+w, y+w) + I(x-h, y+h) - I(x, y) - I(x+w-h, y+w+h) \quad (2.9)$$

où r est défini comme étant un rectangle ayant subi une rotation de 45 degrés ($r = (x, y, w, h)$)

En 2005, Porikli publie un article [77] appliquant la théorie de l’image intégrale aux histogrammes afin d’effectuer de la détection d’objets ainsi que du suivi de ces mêmes objets.

Un point négatif des techniques utilisant l’apprentissage est que la détection est toujours aussi bonne que l’échantillon utilisé. Par exemple, si une

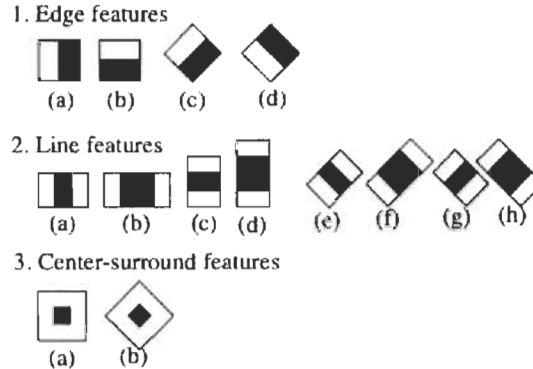


FIGURE 2.3 – Représentation de caractéristiques pseudo-Haar où 1(a), 1(b), 2(a), 2(b), 2(c), 2(d) et 3(a) sont énoncé par [73] et [100] tandis que les caractéristiques 1(c), 1(d), 2(e), 2(f), 2(g), 2(h) et 3(b) sont introduites dans [61] (Tiré de [61])

détection est effectué sur une image contenant des visages à peau noire et que l'apprentissage n'a été fait qu'avec des visages à peau blanche, il est fortement possible que les visages à peau noir ne soient pas détectés. Donc, lors d'un apprentissage, il faut prévoir quels genres de détection seront effectués et prévoir l'échantillon en conséquence.

Une des premières méthodes à être développée et à être brevetée servant à détecter des caractéristiques spécifiques, comme les segments de droite, a été publiée en 1962 par Paul Hough. L'avancée particulière de cette technique a surtout été au niveau du regroupement de points ou de groupe de points dans une image contenant du bruit ou ne présentant pas des lignes continues. Cette théorie part du principe qu'il existe un nombre infini de droites pouvant passer par un même point et ne sont différentes entre elles que par leur angle. Cette théorie permet donc de passer des coordonnées cartésiennes aux coordonnées polaires permettant d'identifier un segment détecté par une norme vectorielle et d'un angle. Les chercheurs Duda et Hart [29] ont poussé la transformation de Hough à un autre niveau en continuant les recherches et en publiant la Transformée Généralisée de Hough qui permet la détection de formes arbitraires. Se basant sur la technique de base de Hough, on recherche la présence d'une courbe passant par un maximum de points détectés et la courbe détectée correspond à celle passant par le plus de points. Un des inconvénients de cette méthode est que plus le nombre de

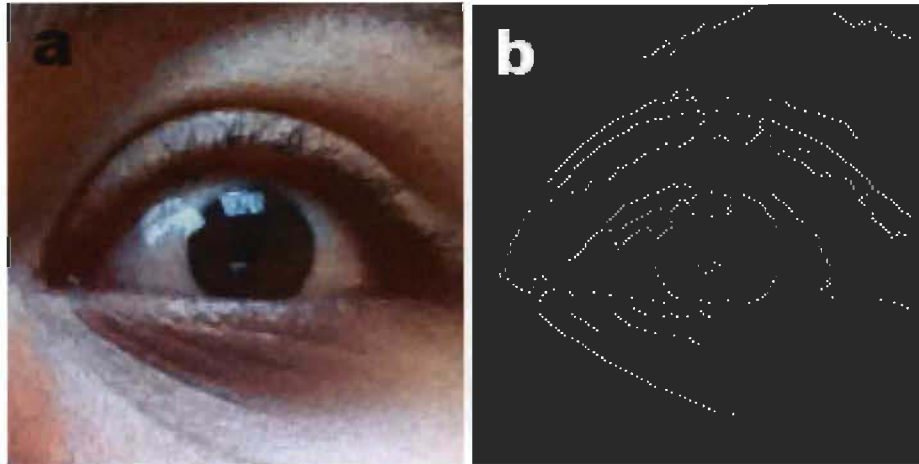


FIGURE 2.4 – Dans l'ordre image originale d'un oeil (a), image de l'oeil ayant subi l'application du filtre de Canny (b)

points est grand, plus l'espace de stockage doit être grand et que le temps requis pour l'analyse s'allonge. Dans le contexte de cette recherche, la détection de formes circulaires est la base de l'algorithme à développer puisque le but ultime est de détecter la présence à la fois de l'iris et de la pupille, qui sont deux cercles concentriques, dans un oeil.

La détection de ligne présentée précédemment est souvent relié à la détection de contours présente dans une image. Selon l'encyclopédie des mathématiques [63], un contour est défini comme étant une discontinuité dans la luminosité d'une image provenant soit d'une discontinuité dans la profondeur, l'orientation de la surface, le reflet ou la luminosité. Une technique très efficace et répandue pour effectuer la détection de contours est l'application du filtre de Canny [11] sur une image. L'application de ce filtre sur une image est présenté à la figure 2.4.

Tel que mentionné précédemment, une fonction très souvent utilisée afin d'effectuer le lissage d'une image est l'application du filtre de Gauss. Ce filtre consiste à effectuer une convolution avec une fonction gaussienne sur une image afin de faire disparaître le bruit d'une image. Ce filtre est l'un des plus utilisés à cause de son efficacité ainsi que par sa rapidité d'exécution.

Bien que le filtre de Gauss soit efficace, d'autres techniques ont été créées

afin d'améliorer le travail du filtre ou de le compléter comme l'application de la théorie de la morphologie mathématique appliquée à l'analyse d'image. Cette théorie a été développée au milieu des années 60 par le travail conjoint de Georges Matheron et de Jean Serra à l'école des Mines de Paris. À l'origine, cette technique a été conçue pour effectuer une classification des minerais selon leur forme. Suite à l'ouverture du Centre de Morphologie Mathématique à Paris en 1968, cette discipline mathématique a connu un développement et d'autres opérateurs morphologiques ont été ajoutés à la technique originale, tels que les opérations d'érosion, de dilatation, d'ouverture et de fermeture, telle que décrite dans cet article de Serra [86].

Opération d'érosion

L'opération d'érosion permet d'amincir une structure géométrique, par exemple un segment de ligne. Il est possible de traiter l'écriture manuscrite, afin d'amincir les traces de crayon à l'aide d'une opération d'érosion. Mathématiquement l'érosion est définie par

$$A \ominus B = \{z \in E \mid B_z \subseteq A\} \quad (2.10)$$

où B_z est la translation de B par un vecteur z tel que $B_z = \{b + z \mid b \in B\} \forall z \in E$. L'érosion est également définie par

$$A \ominus B = \bigcap_{b \in B} A_{-b} \quad (2.11)$$

où A est une image binaire, B est l'élément structurel de l'opération ou « kernel » et E est un espace euclidien. Comme démontré à la figure 2.5, l'opération de dilatation est souvent utilisée afin d'enlever le bruit dans les images, comme dans le cas présent dans l'oeil afin de détecter plus précisément la position de l'iris et de la pupille dans l'oeil.

Opération de dilatation

L'opération de dilatation permet d'épaissir une structure géométrique, tel un segment de ligne. Par exemple, il est possible de traiter l'écriture manuscrite afin d'élargir chacune des traces de crayon pour une meilleure lecture tout en préservant la forme des lettres originales. Mathématiquement, la dilatation est définie par

$$A \oplus B = \bigcup_{b \in B} A_b \quad (2.12)$$

où A est une image binaire et B est l'élément structurel de l'opération ou « kernel ». Cette opération est commutative. Un exemple de cette opération appliquée sur l'image d'un oeil est présentée à la figure 2.5.

Opération d'ouverture

L'opération d'ouverture permet de former les frontières d'une structure géométrique. Par exemple, elle arrondit les coins lors de l'utilisation d'un filtre circulaire. Mathématiquement, l'ouverture est définie par

$$A \circ B = (A \ominus B) \oplus B \quad (2.13)$$

ou

$$A \circ B = \bigcup_{B_x \subseteq A} B_x. \quad (2.14)$$

où A est une image binaire et B est l'élément structurel de l'opération ou « kernel ».

Opération de fermeture

L'opération d'ouverture permet de créer des frontières entre deux structures géométriques. Mathématiquement, la fermeture est définie par

$$A \bullet B = (A \oplus B) \ominus B \quad (2.15)$$

où A est une image binaire et B est l'élément structurel de l'opération ou « kernel ».

Dans le cadre d'une détection d'iris et de pupille, les opérations de morphologie sont souvent utilisées afin de lisser l'image segmentée de l'oeil et de réduire la présence de reflet de lumière sur le globe oculaire afin de mieux détecter l'iris comme présenté sur l'image d'érosion à la figure 2.5. En plus d'aider à localiser l'iris, l'érosion permet également de trouver la pupille, puisqu'elle permet d'éliminer le bruit dans l'image de l'iris et donc de faciliter la détection de cercles concentriques dans une image de l'oeil. Quant à la dilatation, elle sert surtout à reconstituer l'image d'origine suite à une opération d'érosion tel que présenté à la figure 2.5. Grâce aux 2 opérations combinées, il est possible de retrouver l'image ayant sensiblement la même grandeur, mais en présentant moins de bruit, ce qui facilite la détection de formes et de contours.

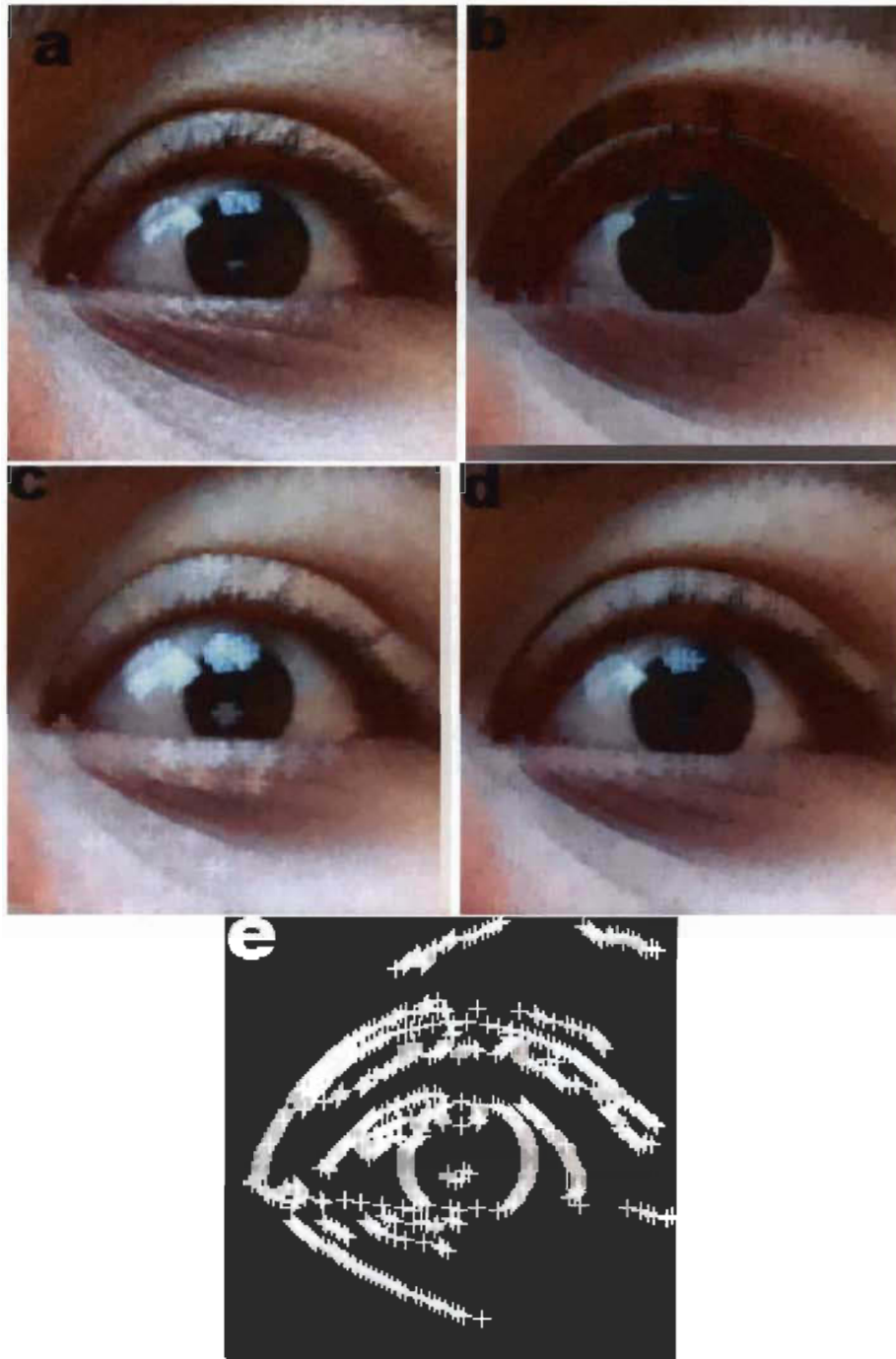


FIGURE 2.5 – Dans l'ordre image originale d'un oeil (a), image de l'oeil ayant subi une opération d'érosion (b), image de l'oeil ayant subi une opération de dilatation (c), image de l'oeil ayant subi une opération d'érosion suivi d'une dilatation (d), image de l'oeil ayant subi un traitement par un filtre de canny et une opération de dilatation par la suite (e).

Finalement, plusieurs articles parlent de la détection d'iris et de pupille comme [49], [48], [93], [76] et [69]. Bien qu'il s'agissent quelques fois de buts différents, la problématique reste la même, soit le fait de trouver deux cercles concentriques entourés d'un oval blanc. Certains chercheurs comme Jan et al. ([49],[48]) utilisent la transformation de Hough, les histogrammes et l'intensité sur une image en niveaux de gris. Par contre, bien que l'on discute de localisation d'iris dans les articles de DeMarsico et al. [64], Tomeo et al. [99] et Rai et al. [79], ces techniques cherchent plus à faire de la reconnaissance de l'empreinte de la rétine que de la localisation, procédé que l'on voit de plus en plus dans les aéroports avec les programmes destinés aux voyageurs fréquents tel que NEXUS[25] et tel que décrit par John Daugman [20].

2.3 Développement mobile

Bien que le développement d'applications mobiles soit devenu une branche du développement d'application en général, cette spécialité a des caractéristiques et des défis bien à elle.

Dans son article [102], Wasserman détaille 8 différences majeures entre le développement d'application standard et le développement mobile. Il s'agit de l'interaction avec d'autres applications, gérer l'interaction avec les différents capteurs sur les appareils, les applications hybrides, les familles d'appareils, la sécurité, l'interface utilisateur, les tests et la consommation d'énergie. Chacune de ces caractéristiques seront discutées dans cette section.

Pour plusieurs entreprises, le passage au développement mobile amène un questionnement profond à savoir si la réutilisation de code existant, et souvent datant de plusieurs années, est pertinent ou doit être réécrit du début. De plus, plusieurs de ces logiciels ont été écrits il y a plusieurs années, sont stables et ne changent que très rarement. Comme le mentionne Rick Leander dans son article [58], le mobile ajoute une pression supplémentaire aux départements de TI provenant des utilisateurs et des clients voulant utiliser de nouvelles façons d'interagir avec leur travail.

Le même questionnement doit également être fait lorsque l'application mobile développée doit rouler sur différentes plateformes. Puisque les différentes plateformes n'utilisent pas la même langage de programmation à la base (Objective-C pour iOS, Java pour Android, .NET/C#/VB pour Windows), il peut être difficile de réutiliser le même code sans avoir à le réécrire en

différents langages. De plus, une autre difficulté est la différence de fonctionnement entre les plateformes. Un exemple pertinent donné dans Joorabchi et al. [50] mentionne les différentes façons que les notifications arrivent aux appareils. Pour iOS, tout passe par les serveurs d'Apple sans que l'application ne soit réveillée, ce qui est tout le contraire pour Android.

Comme l'affirme Wasserman, une application mobile se doit d'interagir avec les autres applications présentes sur l'appareil sur laquelle elle est installée. Que se soit de faire afficher une page web dans le navigateur web de l'appareil ou d'envoyer une notification à l'utilisateur, l'application se doit de respecter les normes de la plateforme et d'être une bonne citoyenne dans son milieu. Il est aussi vrai lorsque vient le temps d'utiliser les données sur l'appareil, comme par exemple le carnet d'adresses. Bien qu'il est acceptable sur un PC d'avoir un carnet d'adresses par application de courriel, il serait très mal vu d'une application de courriel de ne pas se soucier des données de contact sur l'appareil et de seulement se baser sur les siennes.

Selon Wasserman, un avantage d'une application mobile sur une application web est la possibilité de maximiser l'utilisation des fonctionnalités offertes par l'appareil, que ce soit par un composant GPS, une boussole ou d'un module d'authentification par empreinte digitale.

Dans son article, Wasserman désigne par le terme "application hybride" une application mobile qui est en partie sur le web ou qui en partie utilise un navigateur web à même l'application, ce qui était une pratique courante en 2010 lors de l'écriture de cet article. Par contre, en 2014, cette pratique est pratiquement disparue au profit des applications mobiles installées mais qui utilisent un fonctionnement client-serveur. Pour la majorité des applications mobiles connectées, la partie importante de l'écosystème de l'application est le serveur web qui gère les données, les accès et les échanges de données. Une analogie simple pouvant être faite avec l'informatique de bureau est que l'application mobile est le navigateur web dans un échange avec un serveur web ou un client dans une architecture client/serveur. Cette partie souvent oubliée de l'architecture d'une application mobile est souvent la partie la plus importante et est celle qui peut faire la différence entre une application médiocre qui offre aucune interactivité ni aucune nouveauté, versus une application populaire interactive. Comme l'affirme Kin Lane sur le site Internet API Evangelist [55], plusieurs applications ont connu leur popularité avec leur interactivité et leur innovation comme eBay, Facebook, Amazon et Instagram. En plus de leur propre serveur et application, plusieurs compagnies

offrent des interfaces de programmation permettant à certaines personnes d'utiliser leur services afin de bâtir leur propre client ou service à partir de leur données. Un bon exemple de ceci est le réseau social Twitter qui permet à n'importe qui de pouvoir interroger ses serveurs en utilisant son interface de programmation. Il s'agit donc maintenant d'un composant essentiel dans un écosystème d'une application mobile.

Wasserman mentionne également les défis de bâtir une application qui puisse s'exécuter sur tous les appareils d'une même plateforme. Bien qu'habituellement le code fonctionne sur tous les appareils, il est souvent nécessaire de prévoir le comportement d'une interface lorsque l'application est exécutée sur un appareil possédant un écran de 3.5" versus un écran de 5.5" ou sur un appareil de type téléphone versus une tablette roulant tout les deux le même système d'exploitation. De plus, il est souvent de mise de se questionner à savoir si les fonctionnalités offertes sur un type fait du sens sur un autre. Pour le développeur, il s'agit d'avoir à bâtir au minimum deux interfaces utilisateurs différentes pour la même application et chacune de ces interfaces se doit de s'adapter à l'écran sur lequel elle sera affichée et de prendre avantage des fonctionnalités offertes sur un certain appareil haute-gamme (ex. : NFC, LTE, etc.) qui ne sont pas nécessairement offertes sur un appareil bas de gamme.

Dans son article, Wasserman discute de la sécurité au niveau des virus et des applications mouchards et que parce que l'appareil mobile est un système fermé, il est moins vulnérable que son cousin l'ordinateur de bureau. Bien que certains cas de vulnérabilité existent dans la théorie et dans les laboratoires, il a été très rare que des attaques de virus puissent faire autant de dommage et du même type que les virus informatiques destinés aux ordinateurs de bureau. En effet, plusieurs sources en font mention sur iOS ([106], [80]) et sur Android ([34]). Sur cette dernière plateforme, il existe même des applications anti-virus.

Un nouveau concept imaginé depuis plusieurs années, mais qui a vu son introduction dans les appareils mobiles, est celui du bac à sable ou "Sandbox". Ce contexte est simple : l'application est autorisée seulement à accéder à ses propres ressources locales et doit demander la permission au système d'exploitation avant d'utiliser certaines ressources externes comme une connexion réseau ou un fichier. Puisque le système d'exploitation bloque toutes les requêtes de l'application vers le système de fichier qui n'est pas dans son espace réservé ou dans son espace mémoire réservé, il s'agit d'un

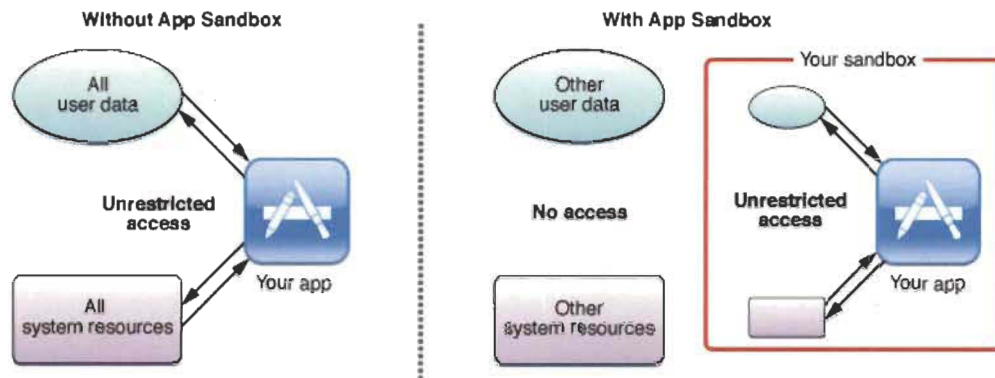


FIGURE 2.6 – Représentation schématique du fonctionnement d’une application régulière et de la même application dans un sandbox. (Tiré de [44])

bon moyen pour garder un système intact et intègre tel que schématisé à la figure 2.6. Par contre, il ajoute souvent des difficultés et des limitations comme exprimées dans cet article.[51]

Dans le contexte de l’application mobile, ce principe est également étendu pour les ressources telles que le carnet d’adresse, les photos ou la caméra, mais au lieu de devoir demander seulement au système d’exploitation, ce dernier demande également la permission à l’utilisateur, ce qui peut rendre une application plus sécuritaire au niveau de la vie privée. Ce niveau de sécurité devient de plus en plus important compte tenu des différents scandales de fuites de photos et d’informations provenant des appareils mobiles d’individus ciblés [104]. Ceci est d’autant plus d’actualité chez le public en général, comme le démontre ce reportage de l’émission La Facture diffusé à Radio-Canada en Octobre 2014[33].

Vu sa caractéristique mobile, l’appareil peut souvent être une source de problèmes du côté sécurité physique, que ce soit par un oubli ou un vol. Que ce soit des courriels, des informations de contacts ou des données sensibles au niveau affaires, il est primordial de s’assurer que les données conservées à l’intérieur soient en sécurité. Pour cette raison, les manufacturiers intègrent de plus en plus de composants matériels pour augmenter la sécurité. Samsung intègre un tel composant dans sa ligne de produit Galaxy [85] et Apple dans ses lignes de produits iPhone et iPad sous la forme de Touch ID [45].

La caractéristique la plus évidente pour un utilisateur d'applications mobiles est son interface utilisateur ainsi que sa facilité d'utilisation. Comme le mentionne Dehlinger et Dixon [24], bien que certains aspects des guides de construction des interfaces utilisateur de chaque plateforme se rejoignent à plusieurs niveaux, plusieurs différences existent également. À cela, Wasserman ajoute que bien qu'il soit utile d'un point de vue du développeur d'avoir une interface universelle sur toutes les plateformes, les utilisateurs d'un certain type d'appareil s'attendent à voir des éléments graphiques semblables d'une application à l'autre sur le même appareil. D'un point de vue du designer, il est souvent nécessaire de repenser au fonctionnement global de l'application lors d'une migration d'une plateforme à une autre, puisque les éléments UI et UX d'une plateforme peuvent permettre des fonctionnalités non-disponibles ou qui ne font aucun sens sur une autre.

Pour ce qui est des tests, Wasserman mentionne surtout les tests à travers les différents réseaux GSM/3G disponibles lors de l'écriture de l'article. Il fait également mention des tests des interfaces utilisateurs face au i18n et aux conséquences d'avoir une application disponible avec différents réglages de langue et de culture. Quant à eux, Joorabchi et al.[50], font mention de tests unitaires, de tests intégrés sous forme de tests automatiques et de tests d'interfaces utilisateurs. Trois raisons principales expliquent ces différences soient l'évolution des réseaux cellulaires du GSM vers le 3G et LTE, l'évolution des outils de développement et des bibliothèques de tests et la maturité des outils et de la discipline. Ils énoncent également plusieurs points intéressants sur les tests des applications mobiles suite à leur sondage avec plusieurs développeurs :

- La culture d'entreprise dans les équipes de développement est beaucoup plus orientée vers la livraison de nouvelles fonctionnalités qu'aux tests.
- Que la présence d'une équipe de tests ou simplement d'une personne dédiée aux tests est très rare.
- Habituellement, les équipes de développement qui travaillent sur diverses plateformes interagissent très rarement entre eux, ce qui forme des tensions et des différences de qualité et de fonctionnalité dans les produits livrés.
- Les plateformes de tests automatiques manquent de fonctionnalités et de fiabilité ce qui ne donne d'autres choix que de se résigner aux tests manuels avec des testeurs.

- Il est impossible de simuler parfaitement la livraison et l'installation de l'application finale via un magasin d'application ce qui a pour effet de créer des situations inattendues lors de la livraison de mises à jour.
- Comme les appareils sont très différents les uns des autres et les systèmes d'exploitation d'une même plateforme évoluent rapidement, il est nécessaire d'exécuter des tests rigoureux sur chacun des appareils et sur les différents systèmes d'exploitation ce qui nécessite un grand nombre d'appareils nécessaires pour tester efficacement.

Donc, à la lumière de ces observations, on s'aperçoit que les tests pour les applications mobiles sont toujours effectués de la même manière que le sont les applications standards.

Finalement, côté performance énergétique, ceci pose de nouveaux défis aux programmeurs et architectes de solution afin d'économiser le plus possible la batterie et de travailler avec les mécanismes en place dans le système d'exploitation pour l'économie de la batterie. Tel que décrit par Speirs [90] dans MacWorld, sur la plateforme iOS, et ce depuis iOS 5, plusieurs applications peuvent être exécutées en même temps, par contre, elles auront quelques limitations. Apple ne permet qu'à 5 types d'applications, dont les applications de Voix-par-IP (VOIP) de rouler en vrai multi-tâche comme une application de bureau. Pour les autres, lorsque l'utilisateur quitte l'application, elle est suspendue et ses tâches sont mises en attente jusqu'à ce que l'utilisateur réouvre l'application. Par exemple, un développeur doit prendre en compte que son application peut être mise en pause d'exécution à tout moment si un utilisateur reçoit un appel sur son téléphone.

Donc, après ce survol de la littérature scientifique dans le domaine de l'analyse des facultés affaiblies, de l'analyse d'image ainsi que du développement d'applications mobiles, il est possible de voir un lien s'établir entre les 3 différents thèmes qui permettrait de bâtir une application mobile permettant d'analyser les captures vidéo d'un individu effectuant un ou plusieurs tests du SFST. Le prochain chapitre présente la méthodologie utilisée afin de développer cette application.

Chapitre 3

Méthodologie et cadre expérimental

Notre problématique de recherche vise à développer une application mobile pouvant détecter la présence de facultés affaiblies chez un individu. Cette application doit se servir de la caméra que contient la plupart de ces appareils afin d'identifier les individus positifs suite à 3 tests distincts permettant d'analyser certains réflexes oculaires à l'aide de séquences vidéos saisies par la caméra. Pour ce faire, nous devons développer les aspects suivants :

- Choisir une plateforme (Combinaison du matériel et du logiciel) afin de développer l'application mobile
- Suite au choix de la plateforme, développer une application mobile facilitant la prise du test par des utilisateurs normaux
- Capturer les séquences vidéos associées aux différents tests afin de déterminer l'état de l'individu testé
- Analyser et afficher les résultats suite à l'analyse des séquences vidéos et la mise en commun des différents tests.

Dans les sections suivantes, nous décrivons plus en détails notre cadre expérimental.

3.1 Choix de la plateforme

Initialement, l'application des techniques développées dans cette recherche devaient être appliquées sur un ordinateur de bureau muni d'une caméra ou

d'un ordinateur de type portable. Par contre, avec l'avènement des nouvelles technologies, ainsi que le développement d'appareils mobiles devenant de plus en plus puissants, la décision a été prise d'explorer l'avenue d'une application mobile.

Considérant les plateformes mobiles présentes sur le marché aujourd'hui, la première chose que l'on a considéré est la partie matérielle de la plateforme, puisque côté logiciel, les plateformes étant dorénavant matures, elles nous permettent d'effectuer notre travail peu importe l'environnement de développement ou le système d'exploitation utilisé. Quelques caractéristiques essentielles doivent être présentes sur l'appareil comme la présence d'une caméra vidéo frontale, ayant une résolution supérieure à 800 X 600, afin d'avoir des images de qualité et la possibilité de développer des applications à l'aide d'outils officiels et supportés ainsi que son utilisation par tout type d'utilisateur, ce qui n'aurait pu être le cas d'une plateforme peu répandue ou complètement développée maison avec un système embarqué.

Afin de pouvoir réaliser le test du nystagmus horizontal, nous avons déterminé que deux caractéristiques sont primordiales soit la présence d'une caméra frontale ainsi que d'une distance considérable (environ 10-15 cm) entre la caméra et les bordures gauche et droite de l'écran. Puisque selon les études, le nystagmus se présente à partir d'une direction de visée de 35 degrés il faut que l'individu puisse être à une distance de la caméra lui permettant de voir le contenu du haut de l'écran et que lorsque l'oeil est centré sur la caméra, le mouvement de la pupille doit être capable de faire un angle de 45° avec le coin gauche ou droit de l'écran tel qu'illustré à la figure 3.1.

Compte tenu de cette caractéristique importante, on doit exclure tous les appareils de type téléphone mobile, ou dit intelligent, à cause de la position générale des caméras frontales vis-à-vis de l'écran. Suite à cette élimination, nous nous sommes retourné uniquement vers les appareils de type tablette. Dans le marché actuel, trois plateformes sont la norme soit le iPad (iPad Air) de Apple, les différentes tablettes de Samsung sur la plateforme Android ou la Surface de Microsoft. Suite à l'obligation que l'oeil forme un angle de 45° avec le coin de l'écran, nous devons éliminer le iPad des choix et la tablette Samsung. La seule tablette restante avec une caméra frontale située selon le sens du format paysage de l'écran est la Surface de Microsoft. Donc, pour ces raisons, c'est cette tablette qui a été choisi pour produire notre application.

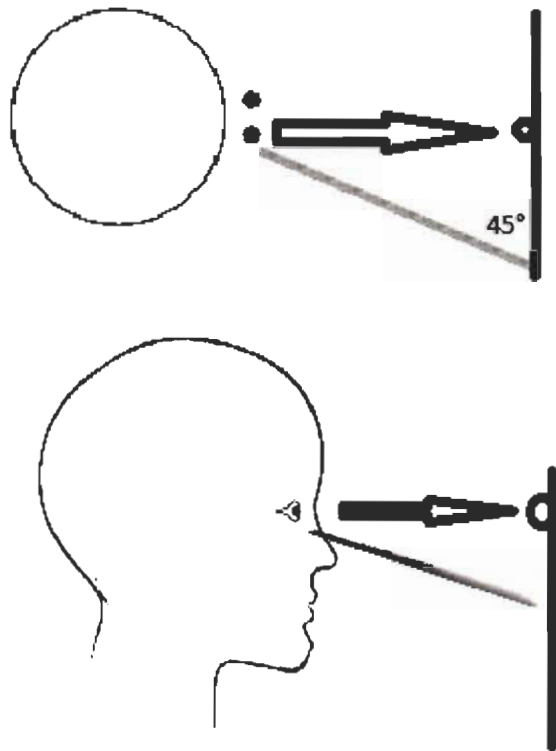


FIGURE 3.1 – Représentation vue du haut (a) et du côté (b) de la position d'un individu fixant la caméra en exécutant le test du Nystagmus.

3.2 Présentation de la plateforme choisie

Annoncée en grande pompe en juin 2012, la tablette Surface de Microsoft est le premier appareil construit par le géant du logiciel. Bien que l'entrée de Microsoft dans le domaine de la construction d'appareil a été une grande nouvelle dans le domaine de l'informatique, l'adaptation de Windows vers le processeur ARM relève d'un exploit. Il s'agissait de la première fois en plus de 10 ans que le noyau de Windows était adapté à une nouvelle architecture et l'entrée de Microsoft dans l'informatique matérielle mobile à partir de la même plateforme qui roule sous des dizaines de millions d'ordinateurs quotidiennement.

3.2.1 Survol de Windows RT et de son architecture

Initialement présentée au CES 2011 sous le nom "Windows On ARM", la plateforme Windows RT est relativement nouvelle chez Microsoft. Présentée comme une version moderne et épurée de Windows roulant sur un processeur 32-bits avec architecture ARM, Windows RT est la riposte de Microsoft envers les autres plateformes mobiles de tablettes telles que iOS pour le iPad de Apple et Android de Google.

Bien qu'au premier coup d'oeil, Windows RT semble semblable à Windows 8 standard, plusieurs différences existent en arrière-plan. Windows RT ne supporte que l'installation d'applications provenant du Windows Store. De ces applications, seules les fonctionnalités de WinRT sont disponibles au programmeur. Les fonctionnalités WinRT sont un sous-ensemble des fonctionnalités standards provenant du .NET Framework et qui rajoute les aspects de sécurité et de "sandboxing". Plusieurs noms ont été données à cet environnement au fil des années soit "Metro", environnement Moderne et RT.

Tout comme le développement d'applications standards sous Windows pour processeurs Intel, le développement d'applications pour Windows RT est effectué à partir du logiciel Visual Studio à partir de la version 2012. Le développement s'y déroule comme une application de style Windows 8 standard. La seule différence notable est l'architecture cible de la compilation qui est pour processeur ARM.

Les caractéristiques détaillées de l'appareil sont présentées à l'annexe 2.

Librairie de traitement d'image

Afin d'effectuer le traitement sur les images capturées grâce au flux vidéo, la librairie libre OpenCV est utilisée. Les raisons principales sont qu'il s'agit de la librairie la plus complète dans le domaine, son coût nul et la communauté dédiée qui développe sans cesse de nouvelles fonctionnalités tout en prenant soin de documenter leurs changements ainsi que d'ajouter des exemples d'utilisation et des articles scientifiques appuyant leur travail. De plus, cette librairie supporte nativement la plateforme Windows RT.

Par contre, une différence notable dans le cas de l'application développée est que les fonctionnalités d'interface graphique et de capture vidéo ne sont pas supportées sous Windows RT ou dans les applications de style Metro sous Windows 8. Afin de contrer cette problématique, le traitement des images tirées des flux vidéos sera effectué via plusieurs composants "Media Foundation Transform" (MFT). L'architecture de ce type de composant est décrite à la prochaine section.

Composants MFT

Les composants MFT sont une évolution directe des objets Direct X Media Objects (DMO) et permettent de faire du traitement vidéo dans une façon moderne et sur demande. Écrit en C++ non géré (C++ unmanaged), le développeur ne peut se servir des notions de .NET, mais doit plutôt s'orienter vers les fonctionnalités Win32. L'avantage d'utiliser du langage compilé (C++ unmanaged) versus l'utilisation d'un langage interprété (.NET/C#) est simple : plus grande vitesse d'exécution du code et moins grande utilisation de la mémoire.

Le principe d'un tel composant est que l'application principale démarre le composant MFT dans un autre processus. Ce fonctionnement permet de faire le traitement du signal ou de l'image de façon asynchrone et donc l'application principale peut continuer de fonctionner normalement sans que l'écran ne soit "gelé" et que le programme soit en attente de réponse. Par contre, dans le cas où le composant MFT doit communiquer avec l'application principale, la situation se corse. Puisque les deux processus sont en situation de "sandbox" ils ne peuvent communiquer directement entre eux. Il serait facile de se tourner vers les standards de la communication interprocessus tel que les pipes, les fichiers ou les sockets, mais ces objets n'existent pas sous WinRT afin de se conformer au "sandbox". Afin de passer outre ce pro-

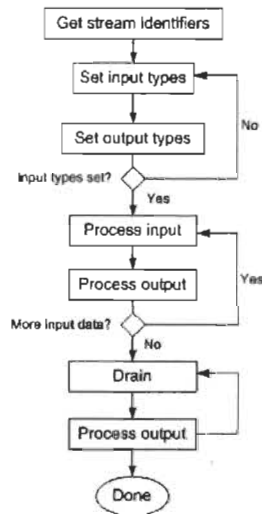


FIGURE 3.2 – Modèle général de traitement d'une MFT

blème, il est nécessaire que l'application principale démarre un serveur web qui peut recevoir des requêtes avec des données et que le composant MFT envoie les données à partir de son traitement d'image. Il s'agit d'un exemple de ce fonctionnement que Microsoft a démontré à la conférence BUILD 2011 [1] et dont le fonctionnement est présenté à la figure 3.2.

3.2.2 Sauvegarde des données

Dans un environnement mobile, les données forment souvent la base de l'application et la raison pourquoi la personne utilise l'application, donc, il est primordial qu'elles soient stockées de façon convenable. De façon standard, les données peuvent être stockées dans un fichier (Texte ou XML), une base de données de base (Tel que SQLite ou SQL Server Express) ou un serveur de base de données relationnelle (Oracle, MySQL, PostgreSQL, MSSQL,...). Par défaut dans une application moderne sous Windows, il est possible de sauvegarder les données au format texte ou XML. Les autres formats mentionnés doivent être implémentés par le programmeur à partir de zéro et ne sont pas disponibles par défaut dans WinRT. Puisque beaucoup de données seront sauvegardées au cours des différentes sessions de détection et que la conservation de l'historique serait une fonctionnalité intéressante, nous devons être en mesure de pouvoir sauvegarder les données d'une façon

structurée et de façon optimale, donc dans une base de données. Deux options s'offraient, soit l'implantation de SQLite et faire les requêtes de bases de données en langage SQL directement sur la base de données ou utiliser une version alpha de Entity Framework (EF) en version 7.0 de Microsoft.

Originellement sortie en 2008, la bibliothèque de fonctionnalités Entity Framework [15] permet de créer une abstraction entre une application et son système de stockage de données. L'avantage de cette bibliothèque est son entière intégration à la bibliothèque .NET Framework et son abstraction du système de gestion de données sur lequel elle travaille. Que les données soient hébergées dans un système de gestion de base de données (SGBD) ou dans un fichier texte, les appels de fonction ne sont pas différents et seule la configuration initiale diffère. Le schéma de la bibliothèque est illustré à la figure 3.3. Au cours des versions, de nombreuses fonctionnalités se sont ajoutées à la bibliothèque comme la possibilité de créer une base de données à partir de classes .NET et l'ingénierie inversée d'une base de données. Une des grandes nouveautés de la version 7.0 est l'ajout du support de Windows RT et de Windows Phone qui était jusqu'ici impossible à faire fonctionner sur ces 2 plateformes.

Grâce aux avantages d'utiliser EF, cette bibliothèque a été choisie afin de stocker les données recueillies lors du traitement des séquences vidéos. Les données seront donc stockées dans une base de données SQLite que EF prendra en charge.

3.3 Algorithmes utilisés pour les différents tests

Afin d'effectuer la détection des facultés affaiblies trois différents tests sont effectués. Les prochaines sections présentent chacun des algorithmes utilisés en détails.

3.3.1 Algorithme général de détection de l'iris

Chacun des trois tests utilise cet algorithme de détection de l'iris dans leur algorithme. Il sera référencé en tant que "Algorithme de détection d'iris" dans les autres sections.

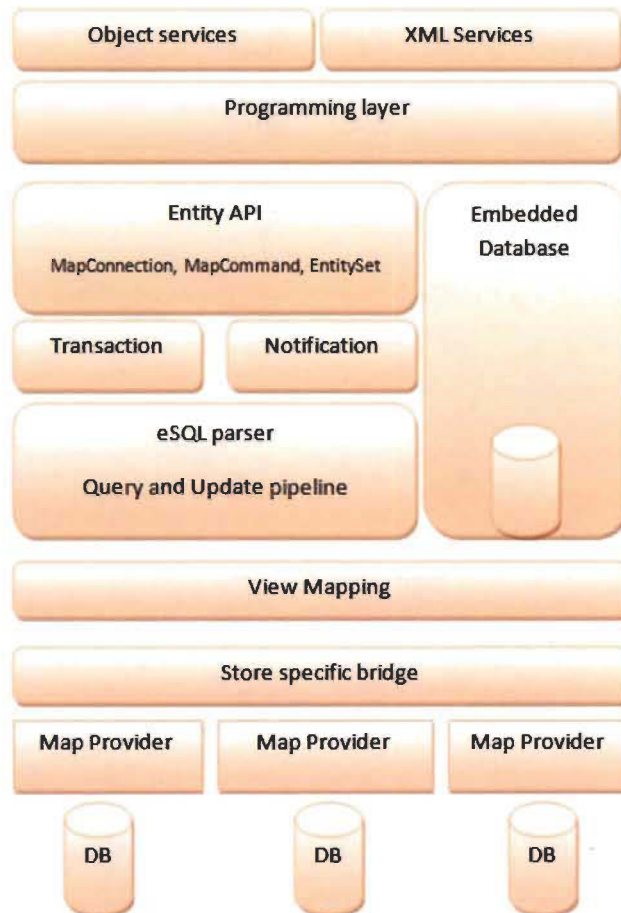



FIGURE 3.3 – *Modèle général du Entity Framework*


algorithme 1 Algorithme de détection de la pupille

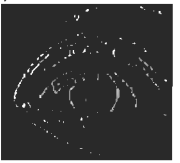
1: **procédure** DETECTIONPUPILLE


2: Image originale 

3: $yeux \leftarrow$ détection d'oeil avec la technique de cascades de haar 


4: **if** 0 == Nombre de yeux **then**
 5: **return** On attends la prochaine image

6: $image1 \leftarrow$ ROI de l'image originale avec l'oeil 
 7: $image2 \leftarrow$ Image en niveaux de gris de l'image 1
 8: $image2 \leftarrow$ Lissage(Image 2)

9: $image2 \leftarrow$ Canny(Image 2) 
 10: $contours \leftarrow$ DetectionContours(Image 2)

11:
 12: **for all** contours **do** 

13: $iris \leftarrow$ PlusGrosCercle(contour, iris)
 14: $imageIris \leftarrow$ ROI(image2, iris)
 15: $imageIris \leftarrow$ Gris(imageIris)
 16: $imageIris \leftarrow$ Lissage(imageIris)

17: $imageIris \leftarrow$ Canny(imageIris) 
 18: $cercles \leftarrow$ DetectionContours(imageIris)

19:
 20: **for all** cercles **do**
 21: $pupille \leftarrow$ CercleConcentrique(cercles, iris)
 22: **return** pupille

3.3.2 Détails des étapes de l'algorithme

Le but de cette section est de détailler certaines étapes de l'algorithme 1 quant aux techniques utilisées et aux images source et de destination que chacune des techniques utilisées.

3. Dans cette étape, la technique des cascades de Haar est utilisée afin de détecter la présence d'un oeil dans l'image. L'entrée est l'image complète du demi-visage capturée par la caméra et la sortie est une liste d'yeux détectés sur l'image.
6. Dans cette étape, l'algorithme définit la région d'intérêt sur l'image à l'oeil détecté à l'étape 3. L'entrée est l'image complète du demi-visage capturée par la caméra et la sortie est un rectangle défini par le contour de l'oeil.
10. Dans cette étape, l'algorithme identifie les cercles potentiels présents dans l'oeil pouvant être des candidats à être identifiés comme étant l'iris de l'oeil. Dans ce cas, l'algorithme de détection des cercles de Hough [41] ne peut être utilisé puisqu'il est possible que l'iris ne soit pas totalement visible (Par exemple, si l'individu n'a pas la paupière totalement ouverte) et il est possible que ce cercle ne soit pas trouvé, ce qui vient faire échouer la détection.
13. Dans cette étape, l'algorithme boucle sur tous les contours trouvés à l'étape 10 et tente de trouver le plus gros cercle situé dans l'oeil. La localisation de ce gros cercle est l'équivalent géométrique de la localisation de l'iris dans l'image.
14. Cette étape permet de définir la région d'intérêt de l'image autour de l'image de l'iris détecté comme étant le plus gros cercle à l'étape précédente.
18. Afin de pouvoir localiser la pupille à l'intérieur de l'iris, il est obligatoire d'identifier les contours présents afin d'éventuellement procéder à la détection de cercles. Pour ce faire, l'algorithme utilise la méthode de détection des contours de [94]. Le résultat obtenu contiendra les contours présents sur l'image et sera ensuite passé à la méthode permettant de détecter les cercles. Dans ce cas, l'algorithme de détection des cercles de Hough [41] à l'aide de la méthode de Yuen [107] aurait également pu être utilisé, mais les résultats en diverses conditions de lumière étaient moins constants que ceux obtenus avec la détection de contours.
21. Dans cette étape, l'algorithme recherche le plus petit cercle, dans la liste de cercles détectés dans l'iris, ayant le même centre que l'iris afin

de détecter la pupille. Si trouvé, on se retrouve en présence de cercles concentriques et il est possible d'affirmer avoir trouvé à la fois l'iris et la pupille. Le centre de la pupille et de l'iris sert par la suite à effectuer le suivi de l'oeil, puisque le centre sera le point de référence utilisé.

3.3.3 Mesure de la variation du diamètre de la pupille par rapport au temps (Réaction au changement de luminosité)

Comme son nom l'indique, le test de réactivité de la pupille à un changement brusque de la lumière consiste, d'abord, à changer rapidement l'éclairage ou la source de lumière afin de passer d'un environnement très sombre vers un environnement très lumineux durant environ 1 seconde et à observer la contraction et/ou la dilatation de la pupille. Dans une situation normale, la pupille devrait se contracter dans un environnement très éclairé et être dilatée dans un environnement sombre et le passage d'un éclairage sombre à clair devrait occasionner une contraction de la pupille et ce dans un délai d'une seconde. Chez les individus intoxiqués, le comportement est souvent aléatoire allant d'une réaction très lente, donc plus longue qu'une seconde, à aucune présence de réaction.

3.3.4 Mesure du diamètre de la pupille en condition de lumière stable (Repos)

Contrairement au test précédent où l'on mesurait la réaction de la pupille dans un environnement sombre et ensuite éclairé, le test, ici, consiste à mesurer le diamètre de la pupille au repos, pendant environ 10 secondes, dans les 2 environnements, sombres et éclairés et à prendre la moyenne des diamètres de la pupille dans chaque environnement afin de les comparer entre elles. Suite à la comparaison, il devrait y avoir une différence significative entre elles.

3.3.5 Test du nystagmus horizontal

Comme décrit précédemment, le test du HGN consiste à analyser le mouvement de la pupille par rapport à un objet qui se déplace. Deux choses sont à analyser pour ce test soit la fluidité du mouvement ainsi que la réaction de l'oeil à un angle de 45°. S'il y a au moins 3 mouvements inconsistants avec le mouvement de l'objet, on considère le test comme un échec.

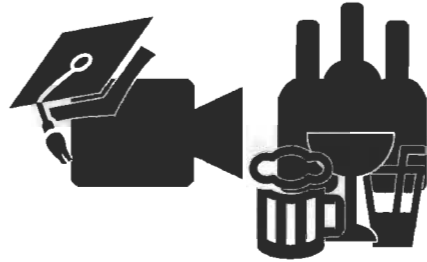


FIGURE 3.4 – Logo de l'application Master

Côté de l'analyse vidéo, on débute par placer un carré au milieu de l'écran (Position 0°) et ensuite on localise la pupille à la position initiale vis-à-vis ce carré. Par la suite, sur chacune des images provenant de la source vidéo, le programme analyse la position de la pupille par rapport au déplacement du carré et ce jusqu'à ce que le carré se place dans le coin de l'écran (Position 45°). À la fin du test, les positions enregistrées sont analysées et le programme détecte la présence d'extremas. S'il y en a 5 ou plus (Position 0° /minimum, 3 pics, Position 45° /maximum), l'application déclare le test échoué).

3.4 Présentation de l'application développée

L'application est intitulée "Master" et représentée par le logo à la figure 3.4 et elle est installée dans les environnements Windows à la figure 3.5. Elle est développée en mode "application universelle". Dans l'écosystème de Microsoft, une application universelle est caractérisée par le fait d'être exécutable autant sur Windows 8 que sur Windows Phone 8.1 à partir du même code de base, tel que représenté à la figure 3.6 par son projet dans Visual Studio. Ceci est un avantage pour les développeurs qui veulent développer sur les deux plateformes tout en réutilisant le même code. Dans notre cas, bien que la version sur Windows soit la principale, il est également intéressant de voir le même code rouler sous Windows Phone sans modification et en développant seulement une nouvelle interface graphique.

L'application permet de fonctionner sous 2 modes, soit l'analyse de vidéo capturé en temps réel ou l'analyse de séquences vidéos déjà préenregistrées. De plus, comme déjà mentionné précédemment, l'application conserve les statistiques d'analyse des sessions précédentes et ces statistiques peuvent



FIGURE 3.5 – L'application *Master* installée sur *Windows RT 8.1* à gauche et *Windows Phone 8.1* à droite

également être consultées à partir de l'écran principal. Afin de faire un test complet, le logiciel propose d'effectuer les 3 tests présentés précédemment soit : le tests du nystagmus horizontal, test de la réaction de la pupille et le test de la variation de la taille de la pupille selon la condition de lumière. Il est également possible de faire seulement 1 ou 2 tests, par contre, avec la possibilité d'un résultat avec moins de fiabilité.

L'écran principal de l'application installée sur une tablette et sur un téléphone est présenté à la figure 3.7. En termes techniques, les deux versions d'applications utilisent exactement le même code en arrière plan, la seule différence vient de l'affichage des différents éléments graphiques de l'interface utilisateur.

L'écran de l'application présenté à la figure 3.8 présente le résultat d'un test de HGN passé avec succès. Les résultats sont disponibles après l'étape d'analyse, à la fin de la batterie de tests, et sont affichés à l'utilisateur à la toute fin et peuvent être également consultés à partir de l'écran principal de l'application.

L'écran de l'application présenté à la figure 3.9 présente l'affichage de l'application lors du test du HGN. Lors de ce test, l'utilisateur doit aligner son



FIGURE 3.6 – Aperçu du projet de base dans Visual Studio 2013. On remarque deux projets distincts (*Master.Windows* et *Master.WindowsPhone*) et la présence d’un projet partagé parent qui regroupe le code et les objets communs aux deux projets enfants. On aperçoit également la présence du composant MFT intitulé *BACDetection*.

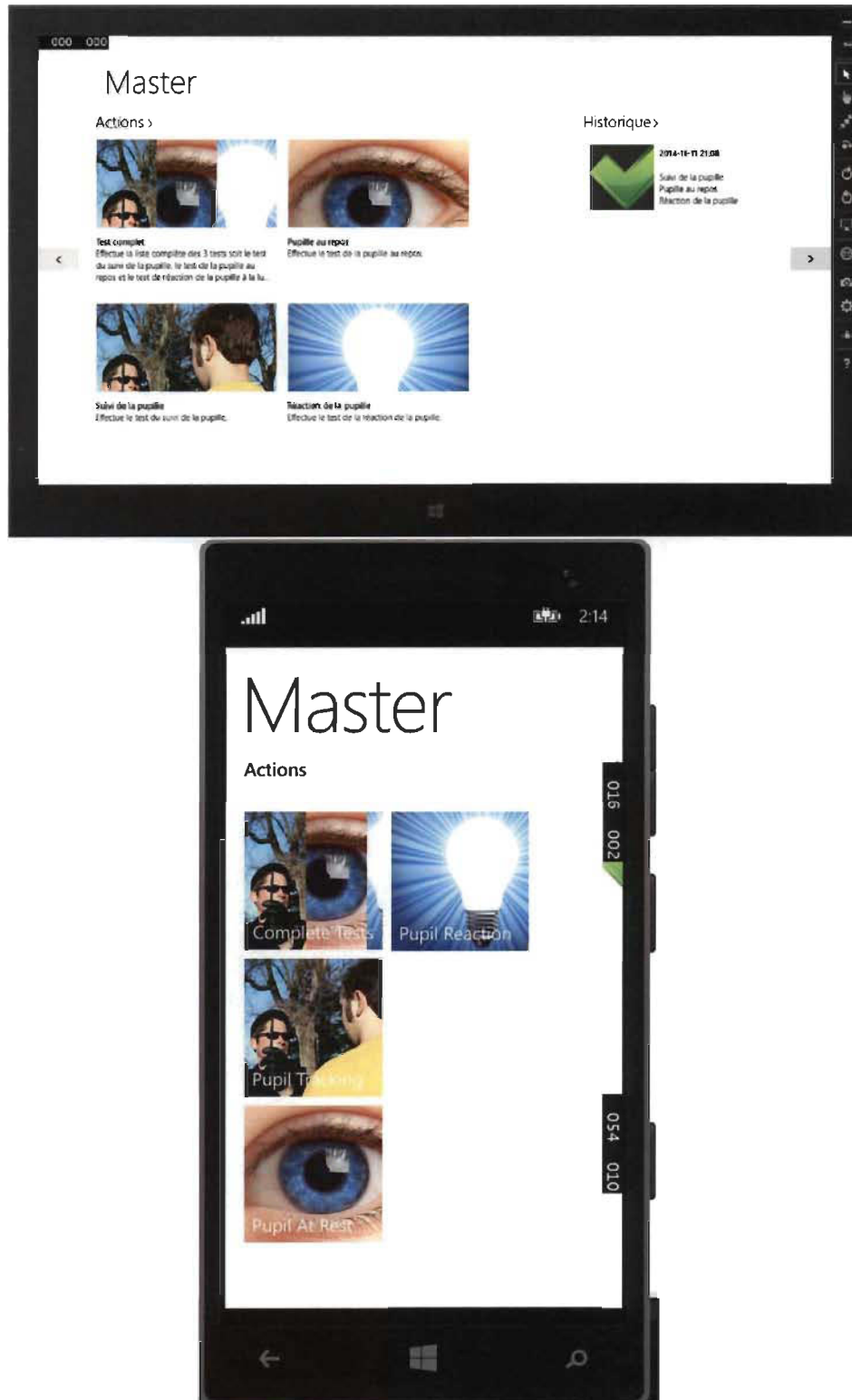


FIGURE 3.7 – Écran principal de l'application Master sur Windows RT 8.1 à gauche et Windows Phone 8.1 à droite



FIGURE 3.8 – Écran du test du HGN de l'application Master sur Windows RT 8.1

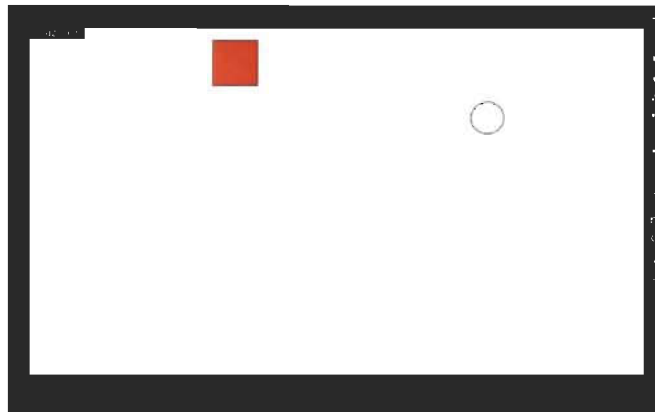


FIGURE 3.9 – Écran de l'application Master durant le test du HGN sur Windows RT 8.1

nez avec le cercle au contour noir affiché sur l'écran et suivre le déplacement du carré rouge sur l'écran avec seulement un oeil, tout en gardant sa tête aligné avec l'écran et son nez vis-à-vis du cercle noir. À la fin du test, l'oeil formera un angle de 45 degrés avec le coin de l'écran.

3.5 Conclusion

En conclusion, ce chapitre a présenté la démarche et les décisions qui ont mené au développement de l'application mobile visé dans cette recherche permettant de détecter la présence de facultés affaiblies chez un individu en analysant divers facteurs des yeux à parir de séquences vidéos. Afin de démontrer le fait d'avoir deux interfaces utilisateur pour 2 plateformes distinctes à partir du même code source, l'application a été développée au format tablette, sur une Surface 2 sur Windows RT 8.1, ainsi que sur un téléphone intelligent, un Lumia 1520 sur Windows Phone 8.1.

Chapitre 4

Résultats et interprétation

Dans un premier temps, nous allons comparer les résultats obtenus à l'aide de l'application versus les résultats de la littérature pour le test du HGN. L'analyse du test se fait en analysant le graphique de la position angulaire de la pupille par rapport au temps. Comme on sait que l'oeil devrait suivre un objet en déplacement constant pendant 4 secondes et qui doit ensuite tenir sa position à la fin du déplacement pour encore 4 secondes, on peut donc procéder à l'analyse de la courbe afin de détecter des minima ou maxima locaux qui nous permettraient de détecter une position non constante ou un déplacement irrégulier de la pupille.

La figure 4.1 présente un exemple de résultat d'un test effectué chez un individu n'ayant consommé aucune boisson alcoolisée. Dans ce graphique, on peut constater que la tendance de la courbe indique que le mouvement de l'oeil est constant avec le déplacement du cube sur l'écran. Les quelques plateaux sur le graphique s'expliquent par le fait d'analyser plusieurs images par secondes du vidéo. Par exemple, dans le cas présenté, les 27 lectures de position ont eu lieu en 6 secondes. Donc, afin de ne pas considérer les plateaux dans les calculs de minima et maxima locaux, il est nécessaire de seulement considérer une lecture sur deux.

La figure 4.2 présente un exemple de résultat d'un test effectué chez un individu (Homme d'environ 185 lbs) ayant consommé et présentant un taux d'alcoolémie aux alentours de 0.08, suite à une consommation de 2 bouteilles de bière à 5% et 3 verres de vin à 11% en 1h30. Sur ce graphique, on peut constater la présence de quelques minima et maxima locaux qui ont pour effet de ne pas produire une courbe lisse dans ce graphique. Par contre, il n'y

a pas un nombre suffisants de minimas et maximas locaux afin de conclure à l'apparition franche du nystagmus horizontal.

La figure 4.3 présente un exemple de résultat d'un test effectué chez un individu (Homme d'environ 210 lbs) ayant consommé et présentant un taux d'alcoolémie aux alentours de 0.12, suite à une consommation de 2 bouteilles de bière à 5%, 4 verres de vin à 11% et 1 shooter de tequila à 40 % en 1h30. Sur ce graphique, on peut constater la présence de plus de 3 minima et maxima locaux qui ont pour effet de ne pas produire une courbe lisse dans ce graphique. Ceci s'explique par la présence du phénomène du nystagmus horizontal qui fait que l'oeil rebondit lors qu'il tend à se déplacer pour former un angle de 45 degrés avec la caméra. Ceci est conforme aux comportements décrits dans les divers articles présentés précédemment.

Pour les tests de mesure de la pupille au repos et la mesure de la dilatation de la pupille suite à un changement d'illumination, les résultats collent également à la littérature. Pour la mesure au repos, l'application mesure le fait d'avoir une différence significative entre la taille devant un écran sombre et devant un écran éclairé, chacune durant environ 4 secondes. Puisque l'application utilise une caméra non-calibrée ainsi que la distance entre la caméra et l'oeil est différente à chacun des tests, il est nécessaire de s'appuyer sur une différence de diamètre relative. Selon [103], la taille de la pupille humaine peut varier entre 2 et 8 mm de diamètre et donc la taille peut varier de 4 fois, donc l'application mesure s'il y a une différence de diamètre d'environ 35 à 50%. Cette mesure a été déduite suite à plusieurs tests afin de trouver l'écart de diamètre optimal étant donné les conditions de lumière que nous offre l'écran de l'appareil. Un exemple de 2 lectures est présenté au tableau 4.1.

	Taille début	Taille Fin	%
Test 1	71	41	42%
Test 2	56	32	43%

TABLE 4.1 – 2 tests de mesures du test de dilatation/contraction de la pupille ainsi que leur pourcentage de variation. Les mesures des diamètres sont exprimés en pixels.

Compte tenu que ces mesures sont obtenues sur des individus n'ayant pris aucune consommation, elles ne reflètent pas le comportement d'un individu

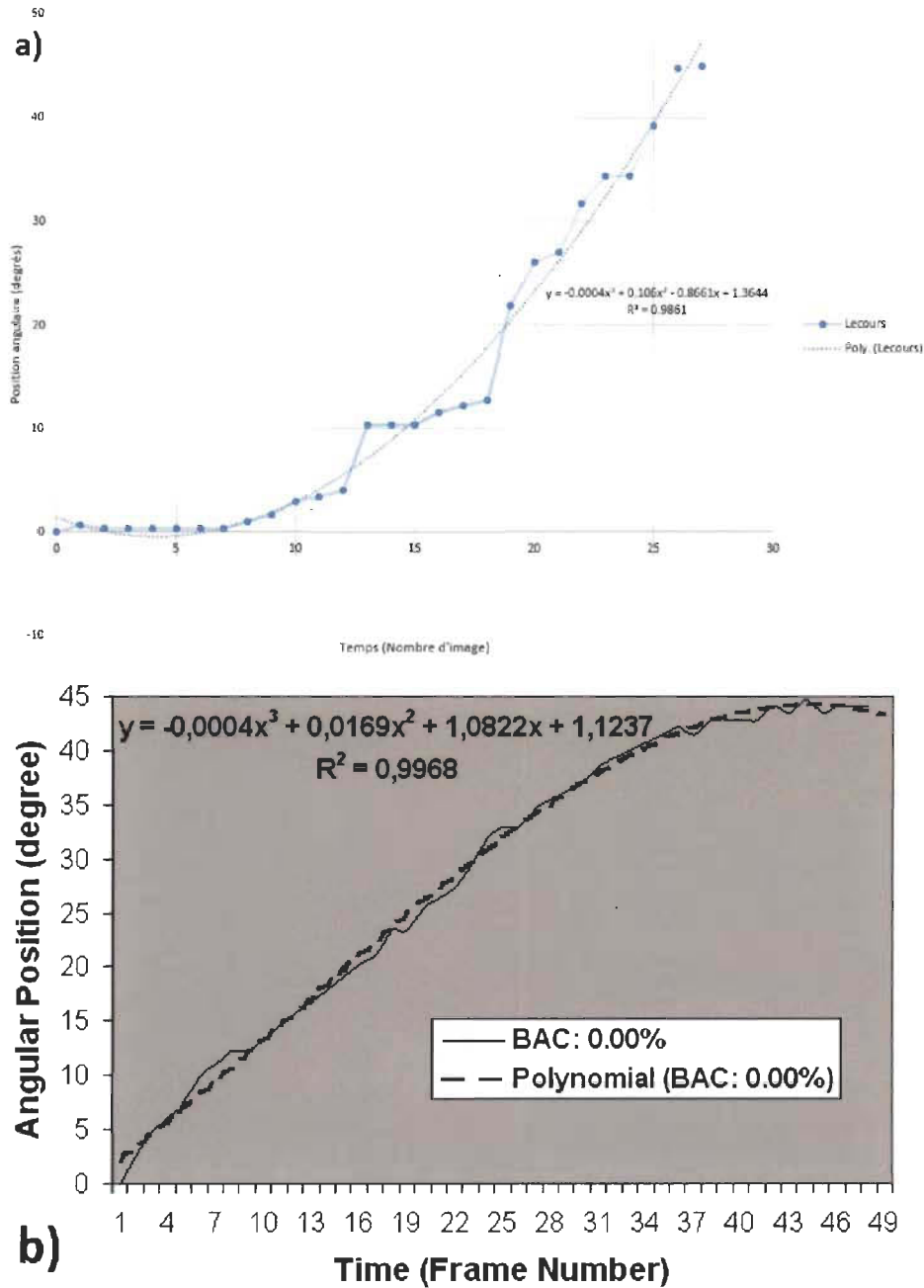


FIGURE 4.1 – Graphique de la position angulaire de l'oeil par rapport à la caméra chez un individu non-intoxiqué. Les résultats obtenus (a) sont comparés avec eux obtenus dans [19] (b) pour un individu avec un pourcentage d'alcool de 0.00%. On peut remarquer la différence de nombres d'images captées pour le test ou (b) a un rythme de 10 images par secondes et (a) environ 4 causé par la charge de travail demandé simultanément à l'appareil par l'application et le système d'exploitation sur d'autres tâches diverses.

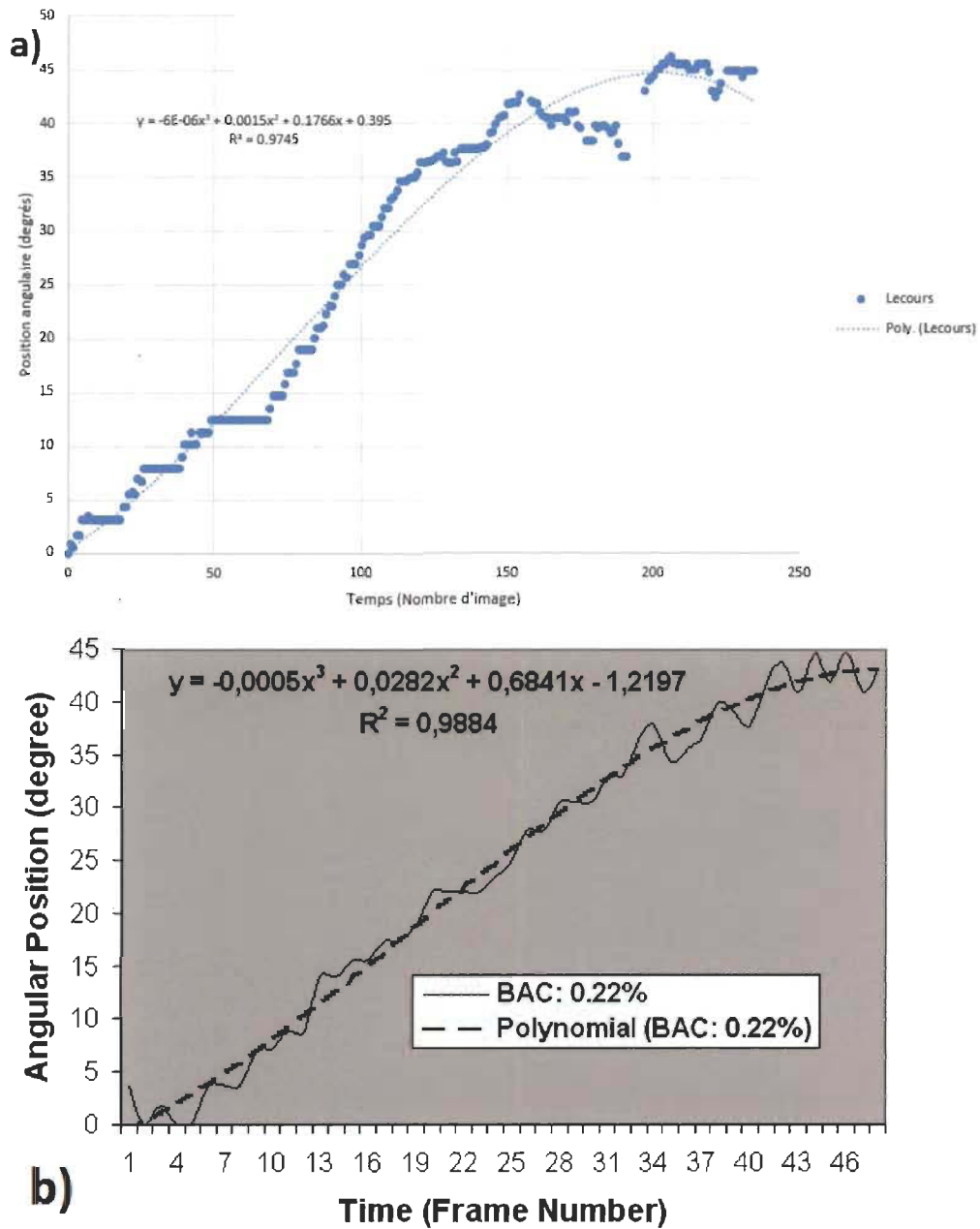


FIGURE 4.2 – Graphique de la position de l'oeil par rapport à la caméra chez un individu commençant à présenter des symptômes de facultés affaiblies avec un taux d'alcool d'environ 0.08%. Les résultats obtenus (a) sont comparés avec eux obtenus dans [19] (b) pour un individu avec un pourcentage d'alcool de 0.09%. On peut observer la même tendance dans les deux expériences malgré des taux de captures différents, mais une durée avoisinant les 8 secondes dans les 2 cas.

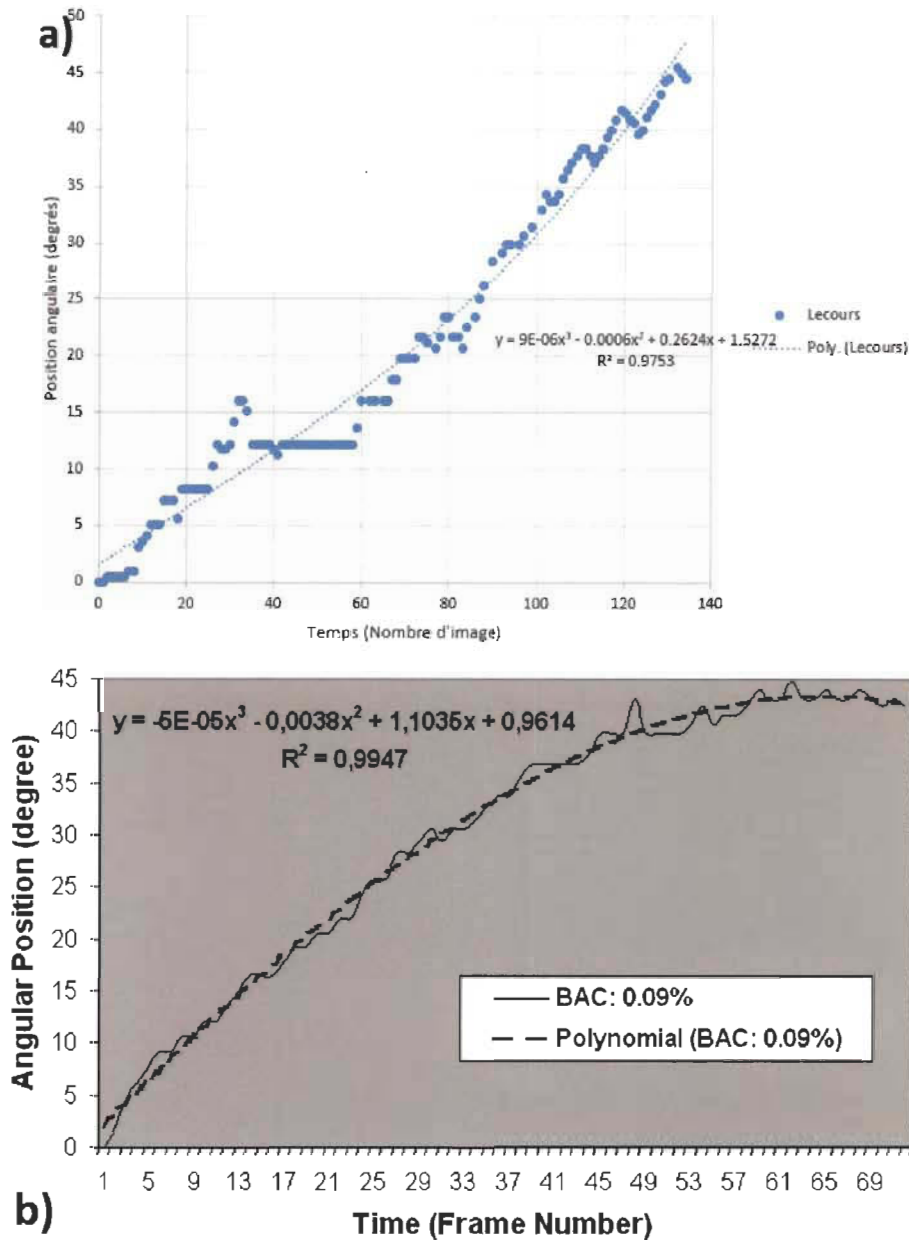


FIGURE 4.3 – Graphique de la position de l’oeil par rapport à la caméra chez un individu présentant des symptômes de facultés affaiblies avec un taux d’alcool d’environ 0.12%. Les résultats obtenus (a) sont comparés avec ceux obtenus dans [19] (b) pour un individu avec un pourcentage d’alcool de 0.22%. Bien qu’il y ait une différence importante entre le pourcentage d’alcool présenté ici et celui par [19], il est possible de voir que la tendance est sensiblement la même, malgré des taux de capture différents, mais une durée sensiblement identique (8 secondes).

lorsqu'il est intoxiqué. Suite à certains tests, il est difficile d'établir un certain standard pour les individus consommateurs, pour 2 facteurs principaux : chacune des substances est différente dans son effet sur le comportement des yeux et il faut aussi considérer le fait que les individus peuvent réagir différemment basé sur plusieurs autres facteurs qui ne sont pas en lien avec la consommation tels que le niveau de fatigue, le moment du jour et la prise de certains médicaments. Dans plusieurs exemples, il est possible de voir que la variation de la pupille entre les deux environnements est négligeable, voire même nulle. Dans certains cas, la pupille reste grande ouverte même en situation d'éclairage maximale et dans d'autres cas, elle reste contractée même en situation d'absence de lumière. Bien que cela ne soit pas une mesure fiable afin de déterminer avec certitude le niveau d'alcool dans le sang d'une personne, il s'agit tout de même d'un bon indicateur sur la santé et les capacités au moment du test d'un individu.

Pour le test de la dilatation de la pupille, qui vise à vérifier le temps de réponse d'une pupille suite à un changement brusque de luminosité, les résultats ont un lien fort sur le test précédent. L'analyse des résultats de ce test consiste à analyser le graphique de la dimension de la pupille par rapport au temps et à vérifier la corrélation avec le changement brusque de la lumière. Selon [43] et [32], le réflexe de contraction de la pupille devrait se compléter en environ 1 seconde, et c'est ce que l'application valide afin de poser son verdict. Comme ce test ne mesure pas la pupille au repos et ne se fiant qu'au changement de diamètre, le test précédent vient le compléter en offrant les deux lectures. Un exemple de résultat est présenté à la figure 4.5 et un exemple de l'image de début de de l'image de la fin du test est également présenté à la figure 4.4.

Donc, bien que les deux derniers tests ne font pas partie des SFST et leur résultat peuvent être aléatoire dans certaines situations, ils sont tout de même de bons indicateurs afin de déceler qu'il y a quelque chose qui cloche chez le comportement d'un individu. Additionné au test du nystagmus horizontal, l'application a une batterie de tests qui lorsqu'exécutés ensembles peuvent donner un diagnostic fiable.

4.1 Discussion

Bien que l'application soit entièrement fonctionnelle, quelques aspects ont soit posé problème, soit apporté un défi supplémentaire durant le déve-

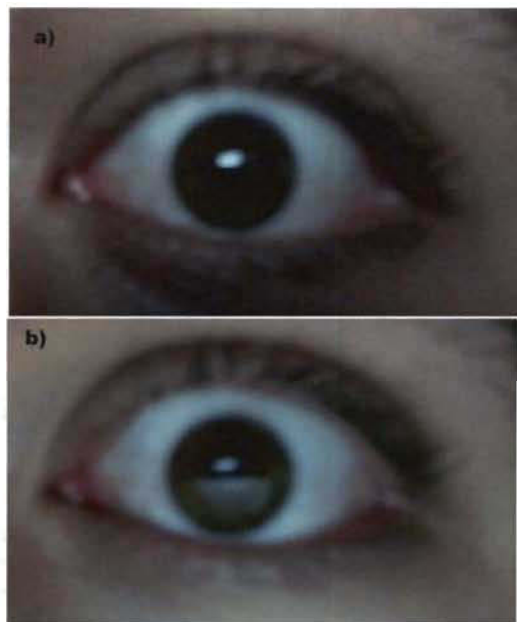


FIGURE 4.4 – Exemples de 2 images qui illustrent les états de départ dans l'obscurité (a) et de fin dans la lumière (b) du test de dilatation/contraction de la pupille.

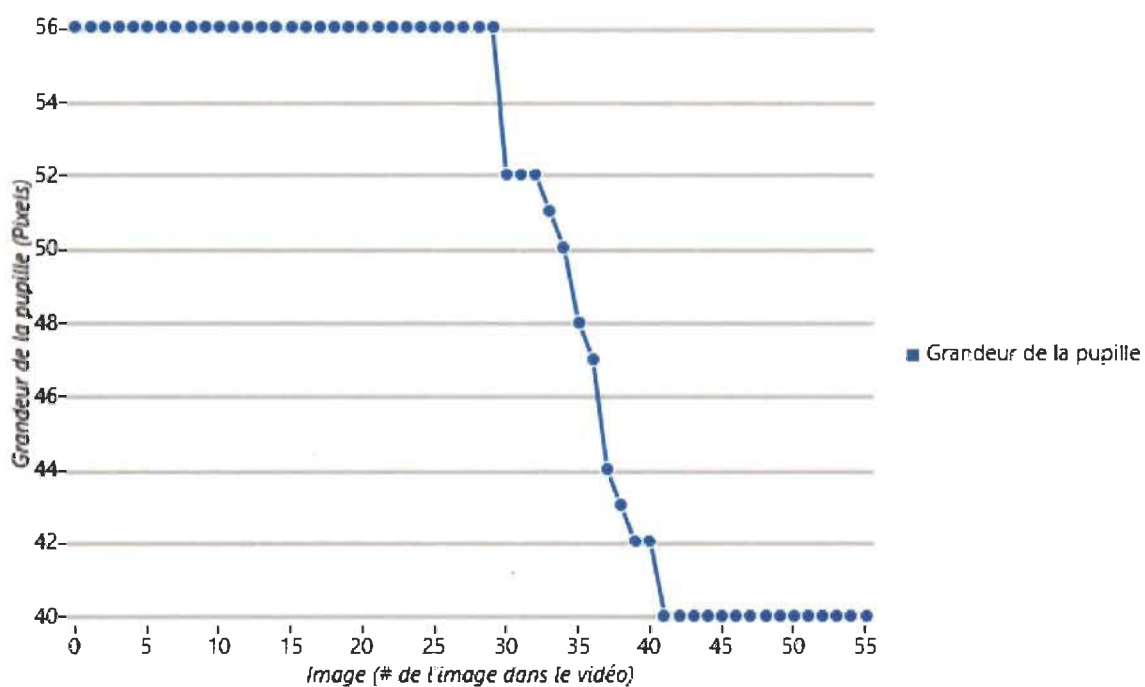


FIGURE 4.5 – Exemple d'un graphique de la variation du diamètre de la pupille par rapport au temps lors d'un changement brusque d'éclairage chez un individu non-intoxiqué. Puisque l'on capture à environ 12 images/seconde et que la réaction s'effectue sur 12 images, le test est passé avec succès par l'individu. Dans le cas présenté ici, le changement de lumière commence à la 30e image du vidéo et se termine à la 42e.



FIGURE 4.6 – Image d'une séquence où un utilisateur passe le test du HGN

loppement et les tests. En voici quelques uns.

Contrairement à la plupart des articles et des applications qui n'analysent que l'oeil, l'analyse des séquences vidéos doit premièrement localiser l'oeil dans une image contenant un fond, un visage et un début de corps tel que démontré dans la figure 4.6. Ceci a pour effet de diminuer de beaucoup la proximité de l'oeil avec la caméra, donc moins de précision dans la détection et la différenciation entre l'iris et la pupille ainsi que de rajouter une étape de plus dans le processus de détection qui consiste à localiser l'oeil sur une image.

À cause de cette distance relativement importante entre l'oeil et la caméra, il devient plus difficile d'obtenir un contraste de couleur important entre l'iris et la pupille dans le cas de gens ayant leur iris foncé, tels que les iris bruns foncés démontrés aux figures 4.6 et 4.7.

Bien que les deux tests du nystagmus présentés précédemment (Figures 4.1 et 4.2) ont été réalisés à partir du même appareil, dans les mêmes conditions ambiantes et ont la même durée, il est possible de voir une grande différence dans le nombre d'images analysées. Ceci s'explique par 2 facteurs dépendants l'un de l'autre, soit la puissance de traitement disponible sur



FIGURE 4.7 – Image d’une séquence où le test se passe dans le noir pour vérifier le temps de réaction de la pupille. On aperçoit également la difficulté à différencier l’iris de couleur brun à la pupille.

la tablette et le fait de faire une capture et une analyse des images capturées en temps réel. Comme l’application utilise un mode de traitement des images asynchrone, avec la librairie MFT de Windows, la capture s’effectue en continue par le système d’exploitation (contrairement à un mode synchrone dont la capture serait gérée par l’application elle-même) et l’application reçoit une image à traiter lorsqu’elle a terminé le traitement sur la dernière image traitée. Par contre, plus le traitement à effectuer est lourd au niveau des ressources matérielles, plus le temps entre les images analysées sera long. Ce phénomène est présent sur la plupart des librairies modernes de capture d’images intégrées aux différents systèmes d’exploitation qu’ils soient mobiles ou de bureau.

Une difficulté rencontrée dans les tests est la difficulté de positionner l’appareil à la bonne distance afin de former un angle de 45 degrés. Lorsqu’un appareil, tel un casque, est totalement dédié à l’analyse du mouvement des yeux, les caméras sont fixées de telle sorte d’avoir toujours la même distance entre chacun des yeux et leur caméra propre. Dans notre cas, l’appareil est mobile ce qui rend cette tâche beaucoup plus difficile. Ce problème peut être contourné en déposant l’appareil sur une surface plane et à faire le test

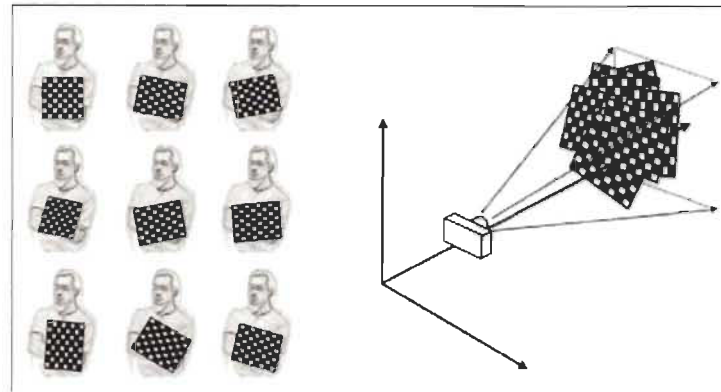


FIGURE 4.8 – Représentation des prises de vues à effectuer à l'aide d'un damier lors de la calibration d'une caméra.

lorsque l'individu est assis et que son visage est placé au dessus de l'appareil.

Une façon de corriger ce dernier problème serait de calibrer la caméra de l'appareil de telle sorte que l'application pourrait être en mesure de déterminer la distance et le positionnement de l'individu et serait en mesure de lui donner des indications de positionnement. La calibration de caméra est un processus qui permet de trouver les caractéristiques d'une caméra (Distance focale, point central, etc.) à partir de plusieurs photos prises d'un objet caractéristique, tel un damier, comme montré à la figure 4.8. Par contre, ceci vient à l'encontre du fait de pouvoir avoir une application facile d'utilisation et pouvant être exécutée sur n'importe quel appareil puisqu'il est virtuellement impossible que toutes les caméras d'appareils d'un même modèle aient exactement les mêmes caractéristiques (Distance focale et point milieu) étant donné leur utilisation initiale prévue.

Le facteur de la luminosité est un facteur important à considérer lors des tests. Premièrement, si les tests de réaction de la pupille sont effectués dans une pièce claire, les tests ne sont pas efficaces, car la pupille est déjà à mi-chemin entre sa dimension dans l'obscurité et sa dimension en milieu éclairé. Par contre, une pièce trop sombre sans aucun autre éclairage a pour effet de ne pas donner assez de lumière à la caméra pour avoir un bon portrait de l'oeil et de la pupille. Il est donc nécessaire d'avoir un compromis entre les 2 soit une pièce pas complètement éclairée, mais qui a une source de lumière d'appoint pas trop loin de la caméra (Éclairage effectué par la luminosité de

l'écran de la tablette). Ce problème est illustré à la figure 4.9

Un problème rencontré lors du développement de l'application a été le fait que puisque la caméra est faite principalement dans le but d'effectuer de la vidéoconférence, elle a un très grand angle, ce qui a pour effet de capter la présence des deux yeux sur l'analyse ainsi que sur les séquences vidéos enregistrées pour le test du HGN. Afin de régler le problème, il est nécessaire de respecter la consigne à l'écran sur l'oeil à tester et l'application se charge de couper la partie d'image qui n'est pas nécessaire afin de ne pas nuire à l'analyse et d'analyser les 2 yeux à la fois.

Bien que l'application développée ici soit fonctionnelle, plusieurs autres ajouts et améliorations pourraient être apportés. La sortie des gadgets intelligents tels que les montres et les appareils permettant de monitorer la santé d'un usager via un appareil mobile sont des avenues intéressantes dans ce sens. Ces appareils permettent entre autre de monitorer le pouls d'une personne en continue et permettent de tenir un suivi dans le temps basé sur les données recueillies lorsque synchronisé à un ordinateur ou à un appareil mobile. Un exemple d'un tel appareil est le bracelet Microsoft Band [16] qui est un hybride entre une montre intelligente et un outil de suivi pour la course. L'apport non-négligeable de ces données mélangé avec d'autres études du milieu médical sur le comportement humain face aux substances ingérés pourraient permettre à l'application d'être plus précise et d'avertir lorsque la consommation devient trop importante pour prendre le volant suite à une analyse des facteurs en temps réel.

En conclusion, suite à l'analyse des résultats, il est maintenant possible d'affirmer qu'il est possible de détecter la présence de facultés affaiblies chez un individu intoxiqué à la suite de l'analyse de séquences vidéos avec une application mobile. Les résultats obtenus montrent une différence notable du comportement d'un individu lorsque qu'il est à jeun et lorsqu'il est sous l'influence de substances soit de l'alcool ou de la drogue ce qui est consistant avec les résultats exposés dans la littérature.

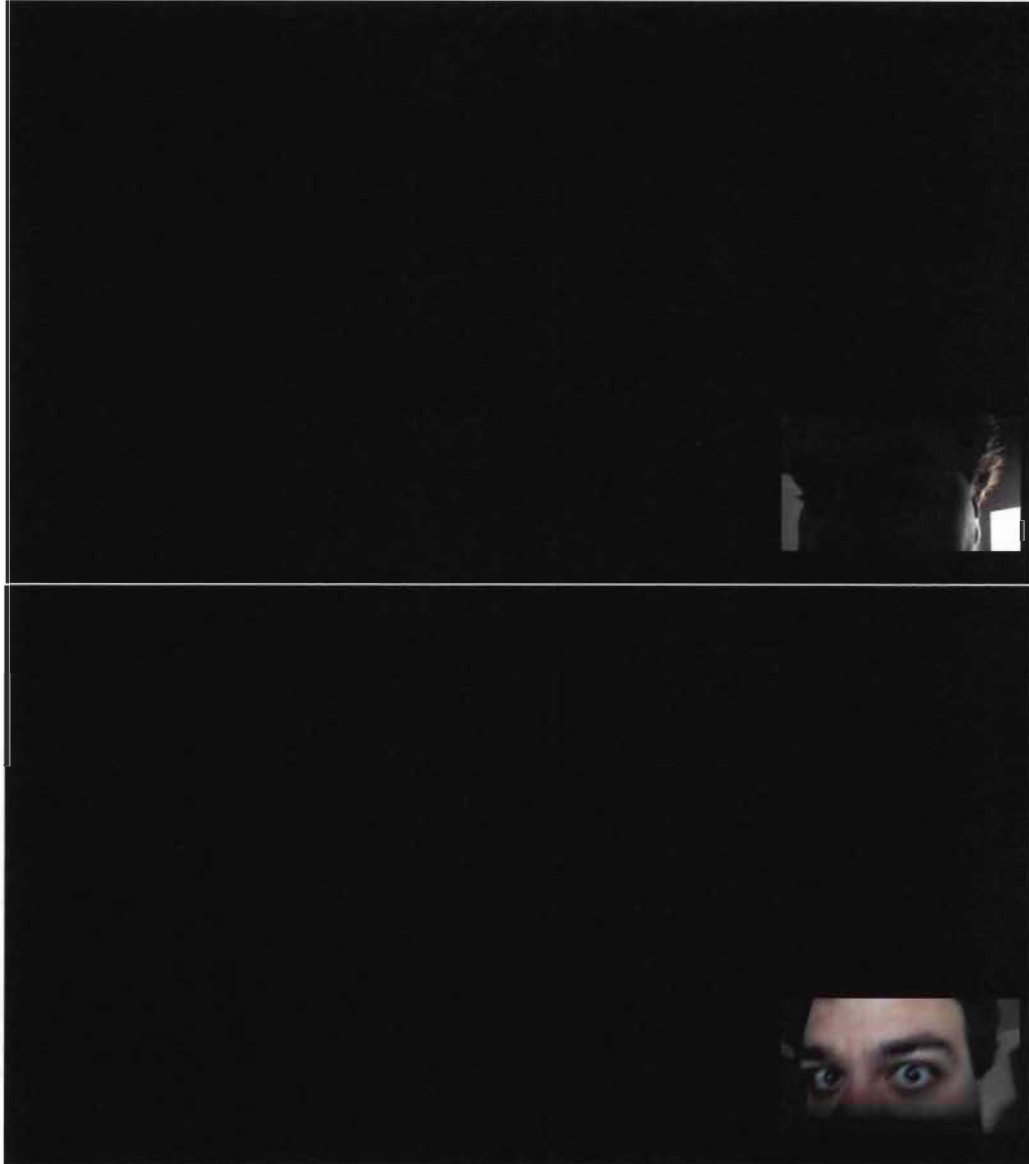


FIGURE 4.9 – Deux captures d'écran de l'application illustrant la nécessité d'avoir un environnement sombre, mais une source de lumière d'ambiance devant l'individu et derrière l'appareil.

Chapitre 5

Conclusion

En conclusion, ce projet de recherche nous a permis de fusionner deux domaines totalement différents soit celui de la physiologie et de l'informatique et qui, à priori, n'ont pas de lien communs forts. Par contre, plus les techniques d'analyses d'images sont développées et plus la qualité, l'efficacité et la fiabilité du matériel informatique et vidéo augmente plus les possibilités de lier des domaines totalement distincts afin de résoudre des problèmes complexes deviennent réelles.

La problématique détaillée de conception d'une application mobile permettant de détecter une personne ayant les facultés affaiblies à partir de l'analyse de plusieurs séquences vidéo et de trois tests distincts est présentée en détail dans le premier chapitre de ce document.

Le second chapitre de ce mémoire présente une introduction générale aux concepts physiologiques de l'humain et du comportement des yeux lorsqu'il y a présence de symptômes de facultés affaiblies, aux différentes techniques d'analyse d'images utilisées dans le processus de conception, ainsi qu'aux caractéristiques et aux problématiques auxquelles sont confrontés les développeurs d'applications mobiles.

La méthodologie développée quant aux différentes techniques d'analyse d'image utilisées ainsi qu'aux principes de conception de logiciel mobile sont décrits dans le 3e chapitre. Ce chapitre présente aussi en détail les décisions prises quant à la conception de l'application mobile.

L'application mobile développée dans le cadre de ce travail est un exemple

des applications mobiles modernes soit une application qui se doit de fonctionner sur différentes plateformes et ayant différentes interfaces utilisateur tout en s'exécutant à partir du même code source. L'aspect multi-plateforme est certes un défi pour les entreprises et les développeurs, mais il permet de rejoindre plus d'utilisateur qu'un produit monoplateforme. Le développement d'un produit permettant de rendre un processus difficile à exécuter est possible grâce aux recherches et aux tests sur le terrain ainsi qu'aux retours d'information du terrain. Ceci permet entre autre à des industries d'augmenter leur efficacité de production avec un meilleur contrôle qualité ou dans le cas qui nous concerne de permettre de pouvoir améliorer le bilan de la sécurité routière en évitant de laisser un conducteur présentant des symptômes de facultés affaiblies sur nos routes.

L'avancée fulgurante des technologies dans les dernières deux décennies permettent aux scientifiques de standardiser plusieurs protocoles et activités auparavant seulement appliqués par un humain. Malgré qu'elle soit le mieux formé possible, le jugement et la perception d'une personne sont toujours présents et font partie de la nature humaine. Les recherches sur le comportement humain dans différents environnement et sous l'influence de différences substances nous aide à comprendre comment le corps humain agit et réagit. Avec la miniaturisation des capteurs et des appareils informatiques, souhaitons que ces avancées technologiques viennent changer pour le mieux notre quotidien et sécurise davantage nos routes et nos sociétés.

Dans la même veine, la dernière année a vu l'apparition de la tendance des application mobiles destinées à la santé. Débutant, avec les applications GPS chronométrant la course et la marche de l'utilisateur nous sommes passés vers des applications prenant avantage des capteurs spécialisés sur les appareils. De plus, avec l'apparition de HealthKit dans la 8e mouture de l'iOS d'Apple et la sortie de montres intelligentes telles la Band de Microsoft et la Apple Watch (à venir en 2015) ayant des capteurs mesurant le sommeil et le pouls, la vague d'applications mobiles liées à la santé est bel et bien démarrée. Dans notre cas, la prise en charge de ces capteurs supplémentaires sur les appareils mobiles pourrait être bénéfique pour améliorer le diagnostic donné par l'application.

Afin d'améliorer la performance de l'application, plusieurs améliorations peuvent être envisagées pour le futur. Côté matériel, afin de réaliser l'analyse vidéo sur plus de caractéristiques et en un meilleur temps, il serait intéressant de se baser sur le meilleur matériel disponible et d'adapter les calculs

sur les nouvelles fonctionnalités des plateformes mobiles telles que le parallélisme et la vectorisation des calculs. De plus, afin d'améliorer le suivi de l'oeil dans l'analyse vidéo, il serait intéressant d'utiliser de nouvelles techniques de détection d'iris et de pupille comme celles publiées par [93] et [95]. Ceci permettrait de ne plus uniquement se fier sur l'échantillon d'apprentissage utilisé et d'améliorer la performance de la détection d'oeil en divers milieux ayant souvent des éclairages différents que ce qui est présent sur l'échantillon.

Bibliographie

- [1] Microsoft Corporation Channel 9. Your metro style app, video and audio, part 2. <https://channel9.msdn.com/Events/Build/BUILD2011/PLAT-776T>, 2011.
- [2] The National Highway Traffic Safety Administration. Development of a standardized field sobriety test (sfst). <http://www.nhtsa.gov/people/injury/alcohol/SFST/>.
- [3] The National Highway Traffic Safety Administration. Standardized field sobriety testing procedures. http://www.nj.gov/oag/hts/downloads/SFST_Procedures.pdf.
- [4] Sunpreet S. Arora, Mayank Vatsa, Richa Singh, and Anil Jain. Iris recognition under alcohol influence : A preliminary study. In Anil K. Jain, Arun Ross, Salil Prabhakar, and Jaihye Kim, editors, *ICB*, pages 336–341. IEEE, 2012.
- [5] Gunnar Aschan. Different types of alcohol nystagmus. *Acta Otolaryngologica*, 49(s140) :69–78, 1958.
- [6] Gunnar Aschan and Martin Bergstedt. Positional alcoholic nystagmus (pan) in man following repeated alcohol doses. *Acta Otolaryngologica*, 79(S330) :15–29, 1975.
- [7] Giovanni Bertolini, Alexander A Tarnutzer, Itsaso Olasagasti, Elham Khojasteh, Konrad P Weber, Christopher J Bockisch, Dominik Straumann, and Sarah Marti. Gaze holding in healthy subjects. *PloS one*, 8(4) :e61389, 2013.
- [8] J.L. Booker. The horizontal gaze nystagmus test : fraudulent science in the american courts. *Science & Justice*, 44(3) :133 – 139, 2004.

- [9] Gary Bradski and Adrian Kaehler. *Learning OpenCV : Computer Vision with the OpenCV Library*. O'Reilly, Cambridge, MA, 2008.
- [10] James L Cambier. Adaptive iris capture in the field. *Biometric Technology Today*, 2014(2) :5 – 7, 2014.
- [11] John Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6) :679–698, June 1986.
- [12] R. Chakravarti and Xiannong Meng. A study of color histogram based image retrieval. In *Information Technology : New Generations, 2009. ITNG '09. Sixth International Conference on*, pages 1323–1328, April 2009.
- [13] T. Praveen Chesti Altaff Hussain, Dr. D. Venkata Rao. Color histogram based image retrieval. In *International Journal of Advanced Engineering Technology*, pages 63–66, Juillet 2013.
- [14] Karl Citek, Ashlee D Elmont, Christopher L Jons, Chad J Krezelok, Joseph D Neron, Timothy A Plummer, and Timothy Tannenbaum. Sleep deprivation does not mimic alcohol intoxication on field sobriety testing*. *Journal of forensic sciences*, 56(5) :1170–1179, 2011.
- [15] Microsoft Corporation. Entity framework (ef) documentation. <http://msdn.microsoft.com/en-ca/data/ee712907>, 2014.
- [16] Microsoft Corporation. Microsoft band - official site. <http://www.microsoft.com/microsoft-band/en-us>, 2014.
- [17] Microsoft Corporation. Surface 2. <http://www.microsoft.com/surface/fr-ca/products/surface-2>, 2014.
- [18] Franklin C. Crow. Summed-area tables for texture mapping. *SIG-GRAPH Comput. Graph.*, 18(3) :207–212, January 1984.
- [19] François Meunier Ph. D. and ing. On the automatic implementation of the eye involuntary reflexes measurements involved in the detection of human liveness and impaired faculties. 2009.
- [20] John Daugman. Iris recognition at airports and border-crossings. http://www.cl.cam.ac.uk/~jgd1000/Iris_Recognition_at_Airports_and_Border-Crossings.pdf.

- [21] Société de l'Assurance Automobile du Québec. Facultés affaiblies - ne conduisez pas. http://www.saaq.gouv.qc.ca/publications/prevention/facultes_affaiblies.pdf.
- [22] Société de l'Assurance Automobile du Québec. Campagne 2011 - "lorsqu'on boit, on ne conduit pas". http://www.saaq.gouv.qc.ca/securite_routiere/comportements/alcool/campagne_2011/index.php, 2011.
- [23] Société de l'Assurance Automobile du Québec. Fatigue au volant. http://www.saaq.gouv.qc.ca/publications/prevention/fatigue_volant.pdf, Octobre 2014.
- [24] Josh Dehlinger and Jeremy Dixon. Mobile application software engineering : Challenges and research directions. In *Workshop on Mobile Software Engineering*, 2011. 25.
- [25] Agence des services frontaliers du Canada. À propos de nexus. <http://www.cbsa-asfc.gc.ca/prog/nexus/about-sujet-eng.html>, Octobre 2014.
- [26] Gendarmerie Royale du Canada. Conduite avec facultés affaiblies chez les jeunes. <http://www.rcmp-grc.gc.ca/cycp-cpcj/id-cfa/index-fra.htm#def>, Octobre 2014.
- [27] Gouvernement du Canada. Evaluation of impaired operation (drugs and alcohol) regulations. <http://laws-lois.justice.gc.ca/eng/regulations/SOR-2008-196/page-1.html>.
- [28] Gouvernement du Québec. Thésaurus de l'activité gouvernementale. <http://www.thesaurus.gouv.qc.ca/tag/acceuil.do>.
- [29] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. Technical Note 36, April 1971.
- [30] Éduc'alcool. Calculateur du taux d'alcoolémie. <http://educalcool.qc.ca/faits-conseil-et-outils/outils/calculateur-du-taux-dalcoolemie/#.VHvZ8uWpr0c>, Novembre 2014.
- [31] Moshe Elzenman, Ping Cheng, James A Sharpe, and Richard C Freccker. End-point nystagmus and ocular drift : an experimental and theoretical study. *Vision research*, 30(6) :863-877, 1990.

- [32] W. R. Biesdorf F. A. Young. Pupillary contraction and dilation in light and darkness. *Journal of Comparative and Physiological Psychology*, (47) :264–268, 1954.
- [33] La Facture. Télécharger une application, un acte de foi! http://ici.radio-canada.ca/emissions/la_facture/2014-2015/Reportage.asp?idDoc=350492, Septembre 2014.
- [34] Joe Fedewa. Android antivirus : 6 truths about smart-phone malware. <http://phandroid.com/2014/05/06/android-virus-malware-scan/>, 2014.
- [35] Andrew Fitzgibbon and Robert B. Fisher. A buyer’s guide to conic fitting. In *In British Machine Vision Conference*, pages 513–522, 1995.
- [36] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1) :119 – 139, 1997.
- [37] Yoav Freund and Robert E. Schapire. A short introduction to boosting, 1999.
- [38] Chris Harris and Mike J. Stephens. A combined corner and edge detector. *Alvey Vision Conference*, pages 147–152, 1988.
- [39] Douglas H. Harris, Robert A. Dick, Steven M. Casey, and Christopher J. Jarosz. The visual detection of driving while intoxicated. 1980.
- [40] H.J.A.M Heijmans and C Ronse. The algebraic basis of mathematical morphology i. dilations and erosions. *Computer Vision, Graphics, and Image Processing*, 50(3) :245 – 295, 1990.
- [41] Paul V.C. Hough. Method and means for recognizing complex patterns. U.S. Patent 3,069,654, Décembre 1962.
- [42] D.E. Howells. Nystagmus as a physical sign in alcoholic intoxication. 1956.
- [43] GRADLE HS and ACKERMAN W. The reaction time of the normal pupil : Second communication. *Journal of the American Medical Association*, 99(16) :1334–1336, 1932.
- [44] Apple Inc. App sandbox design guide : About app sandbox. <https://developer.apple.com/library/mac/>

- documentation/Security/Conceptual/AppSandboxDesignGuide/AboutAppSandbox/AboutAppSandbox.html#//apple_ref/doc/uid/TP40011183-CH1-SW1, Octobre 2014.
- [45] Apple Inc. Apple (canada) - iphone 6 - touch id. <https://www.apple.com/ca/iphone-6/touch-id/>, 2014.
- [46] American Prosecutors Research Institute. *HORIZONTAL GAZE NYSTAGMUS : THE SCIENCE AND THE LAW*.
- [47] Eric J Ip, Quang V Bui, Mitchell J Barnett, Amin Kazani, Roger Wright, Melissa J Serino, and Paul J Perry. The effect of trazodone on standardized field sobriety tests. *Pharmacotherapy : The Journal of Human Pharmacology and Drug Therapy*, 33(4) :369–374, 2013.
- [48] Farmanullah Jan, Imran Usman, and Shahrukh Agha. Reliable iris localization using hough transform, histogram-bisection, and eccentricity. *Signal Processing*, 93(1) :230 – 241, 2013.
- [49] Farmanullah Jan, Imran Usman, Shahid A. Khan, and Shahzad A. Malik. Iris localization based on the hough transform, a radial-gradient operator, and the gray-level intensity. *Optik - International Journal for Light and Electron Optics*, 124(23) :5976 – 5985, 2013.
- [50] M.E. Joorabchi, A. Mesbah, and P. Kruchten. Real challenges in mobile app development. In *Empirical Software Engineering and Measurement, 2013 ACM / IEEE International Symposium on*, pages 15–24, Oct 2013.
- [51] Topher Kessler. What apple’s sandboxing means for developers and users. <http://www.cnet.com/news/what-apples-sandboxing-means-for-developers-and-users/>, 2011.
- [52] Esq. Kurt M. Hughes. A primer on the horizontal gaze nystagmus test.
- [53] Jacques Lacoursière. *Histoire populaire du Québec Tome 4 - 1896 à 1960*, volume Tome 4. Les éditions du Septentrion, 1997.
- [54] Robert Laganière. *OpenCV 2 Computer Vision Application Programming Cookbook*. Packt Publishing, May 2011.

- [55] Kin Lane. Api 101. <https://s3.amazonaws.com/kinlane-productions/whitepapers/API+Evangelist+-+API+101.pdf>, Juillet 2013.
- [56] Ivan Laptev. Improvements of object detection using boosted histograms. In *In Proc. BMVC*, pages 949–958. BMVC, 2006.
- [57] Ivan Laptev. Improving object detection with boosted histograms. *Image Vision Comput.*, 27(5) :535–544, April 2009.
- [58] Rick Leander. Explain the challenges found in developing an enterprise application system. <http://smallbusiness.chron.com/explain-challenges-found-developing-enterprise-application-system-74167.html>, 2014.
- [59] Frederick Leatherman. Horizontal gaze nystagmus (hgn) and the frog gravy legal case. <http://frederickleatherman.com/2011/10/25/horizontal-gaze-nystagmus-hgn-and-the-frog-gravy-legal-case/>, 2011.
- [60] Jessica LeFevre. Horizontal gaze nystagmus : Limiting the admissibility of an unreliable test. <http://www.law.udmercy.edu/udm/images/lawreview/LeFevre.pdf>, 2013.
- [61] Rainer Lienhart, Er Kuranov, and Vadim Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. pages 297–304, 2002.
- [62] Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid objection detection. *IEEE ICIP*, 2002.
- [63] Tony Lindeberg. Edge detection - encyclopedia of mathematics. http://www.encyclopediaofmath.org/index.php?title=Edge_detection&oldid=17883, Octobre 2014.
- [64] Maria De Marsico, Chiara Galdi, Michele Nappi, and Daniel Riccio. Firme : Face and iris recognition for mobile engagement. *Image and Vision Computing*, (0) :-, 2014.
- [65] Francois Meunier and D. Laperriere. A video-based image processing system for the automatic implementation of the eye involuntary reflexes measurements involved in the drug recognition expert (dre). In *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on*, pages 599–605, March 2008.

- [66] The Canadian Press Michael Oliveira. 49 <http://www.cbc.ca/news/business/49-of-time-online-now-spent-on-mobile-devices-comscore-says-1.2794903>, October 2014.
- [67] T. Mita, T. Kaneko, and O. Hori. Joint haar-like features for face detection. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1619–1626 Vol. 2, Oct 2005.
- [68] Microsoft Corporation MSDN. Media foundation transforms. [http://msdn.microsoft.com/en-us/library/windows/desktop/ms703138\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms703138(v=vs.85).aspx), 2014.
- [69] Kien Nguyen, Clinton Fookes, Sridha Sridharan, and Simon Denman. Feature-domain super-resolution for iris recognition. *Computer Vision and Image Understanding*, 117(10) :1526 – 1535, 2013.
- [70] House of Commons. Toward eliminating impaired driving. <http://www.parl.gc.ca/HousePublications/Publication.aspx?DocId=1031597&Language=E&Mode=1&Parl=36&Ses=1#0.2.5FPCNZ.3YC5UE.7ZN1CF.D3>, May 1999.
- [71] Michael O’Keefe. Drugs driving – standardized field sobriety tests : a survey of police surgeons in strathclyde. *Journal of Clinical Forensic Medicine*, (8) :57–65, 2001.
- [72] Nobuyuki Otsu. A Threshold Selection Method from Gray-level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1) :62–66, 1979.
- [73] Constantine P. Papageorgiou, Michael Oren, and Tomaso Poggio. A general framework for object detection. In *Sixth International Conference on*, pages 555–562, 1998.
- [74] Samuel Perreault. La conduite avec facultés affaiblies au canada, 2011. Technical report, Statistique Canada, janvier 2013.
- [75] Karl Citek O.D. Ph.D., Bret Ball O.D., and Lieutenant Dales A. Rutledge. Nystagmus testing in intoxicated individuals. *Optometry*, 74 :695–710, 2003.
- [76] J.K. Pillai, M. Puertas, and R. Chellappa. Cross-sensor iris recognition through kernel learning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(1) :73–85, Jan 2014.

- [77] F. Porikli. Integral histogram : a fast way to extract histograms in cartesian spaces. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 829–836 vol. 1, June 2005.
- [78] Société Radio-Canada. Tester l'alcootest - les archives de radio-canada. <http://archives.radio-canada.ca/emissions/577-11180/page/4/>, Avril 1969.
- [79] Himanshu Rai and Anamika Yadav. Iris recognition using combined support vector machine and hamming distance approach. *Expert Systems with Applications*, 41(2) :588 – 593, 2014.
- [80] Reuters. Advanced ios virus targeting hong kong protestors - security firm. <http://www.reuters.com/article/2014/09/30/hongkong-china-cybersecurity-apple-idUSL2N0RV2D320140930>, 2014.
- [81] Henry A. Rowley, Student Member, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions On Pattern Analysis and Machine intelligence*, 20 :23–38, 1998.
- [82] Steve Rubenzer. Judging intoxication. *Behavioral sciences & the law*, 29(1) :116–137, 2011.
- [83] Steven J. Rubenzer. The standardized field sobriety tests : A review of scientific and legal issues. *Law and Human Behavior*, 32(4) :293–313, 2008.
- [84] Steven J. Rubenzer and Scott B. Stevenson. Horizontal gaze nystagmus : A review of vision science and application issues. *Journal of Forensic Sciences*, 55(2) :394–409, 2010.
- [85] Samsung. Samsung galaxy s5. <http://www.samsung.com/global/microsite/galaxys5/features.html>, 2014.
- [86] Jean Serra. Introduction to mathematical morphology. *Computer Vision, Graphics, and Image Processing*, 35(3) :283 – 305, 1986. Special Section on Mathematical Morphology.
- [87] Mehmet Sezgin and Bülent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *J. Electronic Imaging*, 13(1) :146–168, 2004.

- [88] Josephine Shallo-Hoffmann, Henning Schwarze, HJ Simonsz, and Hermann Mühlendyck. A reexamination of end-point and rebound nystagmus in normals. *Investigative ophthalmology & visual science*, 31(2) :388–392, 1990.
- [89] John A Smith, Charles E Hayes, Robert L Yolton, Dale A Rutledge, and Karl Citek. Drug recognition expert evaluations made using limited data. *Forensic Science International*, 130(2–3) :167 – 173, 2002.
- [90] Fraser Speirs. How ios multitasking really works. http://www.macworld.com/article/1164616/how_ios_multitasking_really_works.html, Janvier 2012.
- [91] Margaret M. Stark, Rob Tunbridge, David Rowe, Paul Fleming, and David Stewart. Drugs, driving and sobriety tests – a review of recent developments. *Journal of Clinical Forensic Medicine*, 9(3) :126 – 132, 2002.
- [92] Jack Stuster and Marcelline Burns. Validation of the standardized field sobriety test battery at bacs below 0.10 percent. <http://ntl.bts.gov/lib/25000/25900/25935/DOT-HS-808-839.pdf>, 1998.
- [93] Shengnan Sun, Shicai Yang, and Lindu Zhao. Noncooperative bovine iris recognition via {SIFT}. *Neurocomputing*, 120(0) :310 – 317, 2013. Image Feature Detection and Description.
- [94] Satoshi Suzuki and Keiichi Abe. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1) :32 – 46, 1985.
- [95] Tieniu Tan, Zhaofeng He, and Zhenan Sun. Efficient and robust segmentation of noisy iris images for non-cooperative iris recognition. *Image Vision Comput.*, 28(2) :223–230, February 2010.
- [96] Sébastien Tempier. La saaq s’attaque à la fatigue au volant. *La Presse*, Juillet 2014.
- [97] MATTHEW J Thurtell and R John Leigh. Nystagmus and saccadic intrusions. *Handb Clin Neurol*, 102 :333–378, 2011.
- [98] Kinh Tieu and Paul Viola. Boosting image retrieval. In *International Journal of Computer Vision*, pages 228–235, 2000.

- [99] Inmaculada Tomeo-Reyes, Judith Liu-Jimenez, Ivan Rubio-Polo, Jorge Redondo-Justo, and Raul Sanchez-Reillo. Input images in iris recognition systems : A case study. In *Systems Conference (SysCon), 2011 IEEE International*, pages 501–505. IEEE, 2011.
- [100] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511–I–518 vol.1, 2001.
- [101] Zhuozheng Wang, Yalei Mei, and Fang Yan. A new web image searching engine by using sift algorithm. *Web Information Systems and Mining, International Conference on*, 0 :366–370, 2009.
- [102] Anthony I. Wasserman. Software engineering issues for mobile application development. In *Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research*, FoSER '10, pages 397–400, New York, NY, USA, 2010. ACM.
- [103] Andrew B. Watson and John I. Yellott. A unified formula for light-adapted pupil size. *Invest. Ophthalmol. Vis. Sci.*, 53(6) :143, 2012.
- [104] Wikipedia. Fuite des photos de personnalités d'août 2014. http://fr.wikipedia.org/wiki/Fuite_des_photos_de_personnalit{é}s_d%27août_2014, Octobre 2014.
- [105] I. M. S. WILKINSON, R. KIME, and M. PURNELL. Alcohol and human eye movement. *Brain*, 97(1) :785–792, 1974.
- [106] Tim Worstall. Apple explains why ios don't need no steenkin' anti-virus. <http://www.forbes.com/sites/timworstall/2012/06/04/apple-explains-why-ios-dont-need-no-steenkin-anti-virus/>, 2012.
- [107] H. K. Yuen, J. Princen, J. Illingworth, and J. Kittler. Comparative study of hough transform methods for circle finding. *Image Vision Comput.*, 8(1) :71–77, February 1990.

Annexe A

Procédure pour administrer le SFST de la NHTSA

**THE NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION
STANDARDIZED FIELD SOBRIETY
TESTING PROCEDURES**

HORIZONTAL GAZE NYSTAGMUS INSTRUCTIONS

1. Please remove your glasses (if worn).
 2. Put your feet together, hands at your side. Keep your head still and look at and follow this stimulus with your eyes only.
 3. Keep looking at the stimulus until told the test is over.
 4. Do not move your head.
 5. Do you understand the directions?
-

WALK AND TURN INSTRUCTIONS

1. Put your left foot on the line, then place your right foot on the line ahead of your left, with the heel of your right foot against the toe of your left foot.
 2. Do not start until I tell you to do so.
 3. Do you understand? (must receive affirmative response)
 4. When I tell you to begin, take 9 heel-to-toe steps on the line (demonstrate) and take 9 heel-to-toe steps back down the line.
 5. When you turn on the ninth step, keep your front foot on the line and turn taking several small steps with the other foot (demonstrate) and take 9 heel-to-toe steps back down the line.
 6. Ensure you look at your feet, count each step out loud, keep your arms at your side, ensure you touch heel-to-toe and do not stop until you have completed the test.
 7. Do you understand the instructions?
 8. You may begin.
 9. If the suspect does not understand some part of the instructions, only the part in which the suspect does not understand should be repeated.
-



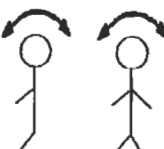

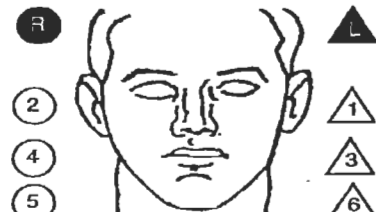
ONE-LEG-STAND INSTRUCTIONS

1. Stand with your feet together and your arms at your side (demonstrate)
2. Maintain position until told otherwise.
3. When I tell you to, I want you to raise one leg, either one, approximately 6 inches off the ground, foot pointed out, both legs straight and look at the elevated foot. Count out loud in the following manner: 1001, 1002, 1003, 1004 and so on until told to stop
4. Do you understand the instructions?
5. You may begin the test.

Annexe B

Questionnaire actuel des symptômes de facultés affaiblies

DRUG INFLUENCE EVALUATION

Evaluator		DRE No.		Rolling Log No.		Session IV	
Recorder/Witness		Crash: <input type="checkbox"/> None <input type="checkbox"/> Fatal <input type="checkbox"/> Injury <input type="checkbox"/> Property		Case #			
Arrestee's Name (Last, First MI)			DOB	Sex	Race	Arresting Officer (Name, ID No.)	
Date Examined/Time/Location				Breath Results: <input type="checkbox"/> Refused Instrument #		Chemical Test <input type="checkbox"/> Refused <input type="checkbox"/> Urine <input type="checkbox"/> Blood	
Miranda Warning Given: <input type="checkbox"/> Yes <input type="checkbox"/> No		What have you eaten today?		When?		What have you been drinking? How much?	
By:						Time of last drink?	
Time now?		When did you last sleep? How long?		Are you sick or injured? <input type="checkbox"/> Yes <input type="checkbox"/> No		Are you diabetic or epileptic? <input type="checkbox"/> Yes <input type="checkbox"/> No	
Do you take insulin? <input type="checkbox"/> Yes <input type="checkbox"/> No		Do you have any physical defects? <input type="checkbox"/> Yes <input type="checkbox"/> No		Are you under the care of a doctor or dentist? <input type="checkbox"/> Yes <input type="checkbox"/> No			
Are you taking any medication or drugs? <input type="checkbox"/> Yes <input type="checkbox"/> No		Attitude:		Coordination:			
		Breath:		Face:			
Speech:		Eyes: <input type="checkbox"/> Reddened Conjunctiva <input type="checkbox"/> Normal <input type="checkbox"/> Bloodshot <input type="checkbox"/> Watery		Blindness: <input type="checkbox"/> None <input type="checkbox"/> Left Eye <input type="checkbox"/> Right Eye		Tracking: <input type="checkbox"/> Equal <input type="checkbox"/> Unequal	
Corrective lens: <input type="checkbox"/> None <input type="checkbox"/> Glasses <input type="checkbox"/> Contacts, if so <input type="checkbox"/> Hard <input type="checkbox"/> Soft		Pupil size: <input type="checkbox"/> Equal <input type="checkbox"/> Unequal, (explain)		Able to follow stimulus: <input type="checkbox"/> Yes <input type="checkbox"/> No		Eyelids: <input type="checkbox"/> Normal <input type="checkbox"/> Droopy	
Pulse and time 1. <u> </u> / <u> </u> 2. <u> </u> / <u> </u> 3. <u> </u> / <u> </u>		HGN Lack of smooth pursuit Maximum deviation Angle of onset		Left Eye	Right Eye	Vertical Nystagnus <input type="checkbox"/> Yes <input type="checkbox"/> No	
						Convergence 	
						One Leg Stand 	
Romberg Balance 		Walk and Turn test 		Cannot keep balance			
				Starts too soon:			
				Stops walking	1 st Nine	2 nd Nine	L R
				Misses heel to toe			<input type="checkbox"/> Sways while balancing
				Steps off line			<input type="checkbox"/> Uses arms to balance
				Raises arms			<input type="checkbox"/> Hopping
				Actual # steps			<input type="checkbox"/> Puts foot down
						Type of footwear:	
Internal clock Est. as 30 seconds		Describe Turn		Cannot do test (explain)		Nasal area:	
Draw lines to spots touched 		Pupil Size	Room Light	Darkness	Direct	Oral cavity:	
		Left					
		Right					
		Hippus: <input type="checkbox"/> Yes <input type="checkbox"/> No		Rebound dilation <input type="checkbox"/> Yes <input type="checkbox"/> No		Reaction to Light:	
Blood pressure: <u> </u> / <u> </u>		Temperature: <u> </u> °F					
Muscle tone: <input type="checkbox"/> Near normal <input type="checkbox"/> Flaccid <input type="checkbox"/> Rigid		Comments:					
What medication or drug have you been using? How much?				Time of use?		Where were the drugs used? (location)	
Date/Time of Arrest:		Time DRE Notified		Evaluation Start Time		Time Completed	
DRE signature (Include rank)		ID #		Reviewed by:			
Opinion of evaluator:		<input type="checkbox"/> Rule Out	<input type="checkbox"/> Alcohol	<input type="checkbox"/> CNS Stimulant	<input type="checkbox"/> Dissociative Anesthetic	<input type="checkbox"/> Inhalant	<input type="checkbox"/> Medical
		<input type="checkbox"/> CNS Depressant	<input type="checkbox"/> Hallucinogen	<input type="checkbox"/> Narcotic Analgesic	<input type="checkbox"/> Cannabis		

Annexe C

Spécifications techniques de la Surface 2 de Microsoft



Surface 2

Système d'exploitation	Windows RT 8.1; Microsoft Office 2013 RT (Outlook, Word, PowerPoint, Excel et OneNote). Fonctionne exclusivement avec les applications disponibles dans Windows Store.
Extérieur	274 x 172 x 9 mm Moins de 675 g Boîtier VaporMg Magnesium (argent) Boutons Volume et Alimentation
Stockage	32 Go*, 64 Go* *Le logiciel système utilise un espace de stockage significatif. Le stockage disponible est susceptible de varier selon les mises à jour du logiciel système et l'utilisation d'applications. 1 Go = 1 milliard d'octets. Reportez-vous à Surface.com/storage pour plus de détails.
Écran	Écran Full HD 10,6 po ClearType 1920 x 1080 pixels 16:9 (écran large) Tactile multipoint 5 points
UC	NVIDIA Tegra 4 1.7GHz Quad Core RAM de 2 Go
Sans fil	Wi-Fi (802.11a/b/g/n) Technologie Bluetooth 4.0
Batterie	Jusqu'à 10 heures de lecture vidéo
Caméras et A/V	Frontale 3,5 MP, arrière 5 MP Deux microphones (avec réduction du bruit) Haut-parleurs stéréo
Ports	USB 3.0 standard Lecteur de cartes microSDXC Prise de casque Port de sortie vidéo HD Port du clavier Cover
Capteurs	Capteur de luminosité ambiante Accéléromètre Gyroscope Magnétomètre
Alimentation	Bloc d'alimentation 24W
Garantie	Garantie matérielle limitée d'un an
Applications (incluses)	Microsoft Office 2013 RT ¹ (Outlook, Word, PowerPoint, Excel et OneNote); Courrier, Calendrier, Contacts, Internet Explorer 11, Photos, Caméra, Skype, SkyDrive, Music, Vidéo, Jeux, etc.

¹ Certaines fonctionnalités et certains programmes ne sont pas pris en charge. Reportez-vous à <http://office.com/officeRT>.

Glossaire

Direct X Collection de bibliothèques permettant le développement d'effets visuels, d'application multimédia et de jeux sous les plateformes Microsoft tel que Windows et Xbox. 29

GSM Abréviation utilisée pour l'expression "Groupe Spécial Mobile" qui désigne les réseaux de télécommunication cellulaire de deuxième génération. 23

HGN Abréviation utilisé pour le terme anglais "Horizontal Gaze Nystagmus" qui indique le phénomène du nystagmus horizontal sur le mouvement de la pupille. 7

i18n Abréviation utilisé pour le terme internationalisation qui dans le contexte de l'informatique désigne l'affichage d'une interface graphique dans différentes langues et différentes cultures. 23

LTE Abréviation utilisé pour le terme anglais "Long Term Evolution" qui désigne un standard de communication pour appareils mobiles implémenter à travers le globe par les opérateurs mobiles permettant des vitesses de transmission de données allant jusqu'à 299.6 Mbit/s en amont et jusqu'à 75.4 Mbit/s en aval. 21, 23

NFC Abréviation utilisé pour le terme anglais "Near Field Communication" qui désigne une technologie de communication sans-fil à courte distance servant majoritairement aux paiements mobiles ainsi qu'à compléter les fonctionnalités du protocole Bluetooth. 21

ROI Abréviation utilisé pour le terme anglais "Region Of Interest" qui indique la région d'intérêt dans une image. 33

UI Abréviation utilisé pour le terme anglais "user interface" qui désigne l'interface utilisateur d'une application informatique.. 22

UX Abréviation utilisé pour le terme anglais "user experience" qui désigne les éléments pouvant influencer le comportement d'un utilisateur avec une application informatique. Ceci inclut également l'analyse des moyens d'interaction entre un individu et une application et/ou un appareil. 22

Visual Studio Nom de l'environnement de développement intégré (IDE) utilisé pour développer des applications sous l'environnement Windows développé par Microsoft. v, 38

VOIP Abréviation utilisé pour le terme anglais "Voice-over-IP" qui désigne les nouvelles technologies liés à la transmission de la voix par le réseau IP. 24

Windows Phone Plateforme mobile pour téléphone de Microsoft. Au moment d'écrire ce mémoire, elle est à la version 8.1. 31