



DESS HANDI

Nouvelles Technologies et Handicaps

Sensori-moteurs

« Analyse et développement d'un logiciel
d'aide à la vision »

CHOQUET Thomas

Directeur de stage : M. Joseph COLINEAU
Entreprise : THALES
Lieu du stage : Palaiseau

Coordonnateur:
J. LOPEZ KRAHE



Paris, Septembre 2005

SOCRATES *Community action programme
in the field of education*

AVERTISSEMENTS

1.1 Liste de diffusion :

Ce document est diffusé en 5 exemplaires.

Trois exemplaires seront remis à l'Université Paris 6, à Mr Jaime LOPEZ KRAHE, directeur du DESS Handi, nouvelles technologies et handicap sensoriels et physiques de l'université Paris VIII.

Un exemplaire sera remis respectivement au maître de stage de l'entreprise d'accueil (THALES Recherche & Technologie) : M. Joseph COLINEAU, ainsi qu'au directeur du service de la Mission Insertion : M. Gérard LEFRANC.

1.2 Reproduction et publication :

Toute reproduction ou publication d'éléments du rapport doit faire l'objet d'un accord préalable écrit par la société THALES.

TABLE DES MATIERES

Pages

INTRODUCTION	8
RAPPEL DU SUJET DE STAGE	9
3.THALES	11
3.1LE GROUPE	11
3.2THALES R&T	11
3.3SERVICE MISSION INSERTION	12
3.4MATERIEL MIS A MA DISPOSITION	14
4.UTILISATEURS FINAUX	16
4.1DEFINITION LEGAL.....	16
4.2CHIFFRE SUR LA POPULATION DEFICIENT VISUELLE	16
4.3TYPES DE DEFICIENCES	17
4.4RAPPORT AUX AIDES TECHNIQUES	20
5.PORTANUM™, VERSION MONOPOSTE	21
5.1DESCRIPTION DU LOGICIEL	21
5.2ACCESSIBILITE ET ERGONOMIE LOGICIELLE	23
5.2.1Simplification de la barre de menu	23
5.2.2Raccourcis clavier.....	25
5.2.3Développement d'une interface.....	27
5.2.4Retours visuels et sonores	28
5.2.5Couleurs, contrastes et police	30
5.3SIMPLIFICATION DE L'ARCHITECTURE DES FONCTIONS DE TRAITEMENT	32
6.PORTANUM™, VERSION RESEAU	37
6.1ANALYSE DES BESOINS	37
6.1.1Vue globale	38
6.1.2Choix du protocole de transport : TCP ou UDP.....	41
6.1.2.1Débit.....	41
6.1.2.2PER : Packet Error Rate.....	44
6.1.3Evaluation des composants Indy (flux multicast UDP).....	46
Expérience 1 : Réception multithread	47
Expérience 2 : Rafraîchissement d'images en fonction de la taille des messages UDP.....	47
Expérience 3 : Perte de paquets dans différentes situations	49
Remarque sur les expérimentations	51
6.2SPECIFICATION	52
6.2.1Réseaux	52
Mode rapide	53
Mode sécurisé	54
6.2.2Traitement et compression des images.....	55
6.2.3IHM	56
6.2.3.1IHM serveur	57
6.2.3.2IHM client.....	57
6.3ARCHITECTURE.....	59
6.3.1Emission.....	59
6.3.1.1Cadencement	59
6.3.1.2Modes d'émission	59
6.3.2Réception.....	60
Réception d'un message UDP.....	60
Gestion de la fluidité.....	61
6.3.3Compression et décompression.....	63

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Côté Emetteur	63
Côté récepteur	64
6.4 VERSION ACTUELLE.....	67
6.4.1 <i>Interface graphique</i>	67
6.4.2 <i>Résultats d'expérimentations</i>	70
Performance de la version monoposte	70
Expérimentation sur la version réseau	71
Autonomie de la nouvelle version.....	73
6.5 DEPLOIEMENT D'UN RESEAU WI-FI.....	75
6.5.1 <i>Portée</i>	75
6.5.2 <i>Interférences</i>	76
CONCLUSION	78
REMERCIEMENTS	79
BIBLIOGRAPHIE.....	80

RESUME

Le groupe THALES a développé « PortaNum™ » : un logiciel d'aide à la vision de loin et de près destiné, principalement, à une population de personnes malvoyantes. Cet outil peut servir dans la vie de tous les jours. En effet, il traite des images afin de les adapter aux préférences de vision de chaque personne. Ces images, capturées à partir d'une Webcam ou d'une caméra numérique, sont retouchées selon des fonctions de traitement d'images à savoir la mise au point, ainsi que la navigation en temps réel de l'image, l'amélioration des contrastes, le renforcement des contours. Tous ces traitements d'images ont comme finalité de favoriser l'intégration de personnes en situation de handicap visuel. Ce logiciel est gratuit et en libre accès sur le site www.PortaNum.com .

En France, même si PortaNum est déjà utilisé par plusieurs personnes malvoyantes, les utilisateurs ont su mettre en exergue le besoin d'améliorer l'interface homme machine afin de la simplifier, de la rendre plus accessible et facile à utiliser dans la vie de tous les jours. De plus, les résultats positifs, du point de vue des traitements d'image et de l'aide que PortaNum apporte aux enfants déficients visuels, ont poussé au développement d'une solution réseau. Grâce à cette dernière, l'utilisation d'une seule caméra sera nécessaire. Elle enverra des images aux différentes machines du réseau permettant aux enfants de travailler sur une image de très bonne qualité et parfaitement cadrée aux dimensions du tableau.

Afin de faciliter l'utilisation de ce logiciel, nous avons décidé de développer une version du logiciel parfaitement accessible en suivant les points les plus cohérents des différentes recommandations. Cette nouvelle version sera ensuite mise en réseau et finira par être installée dès la rentrée prochaine dans les écoles en partenariat avec THALES.

MOTS-CLES

Aide technique, malvoyants, lecture, nouvelles technologies, IHM, traitement numérique d'images, accessibilité, logiciel, réseau, Wi-Fi , 802.11, caméra numérique, déploiement, multicast, UDP, THALES, PortaNum.

ABSTRACT

THALES developed "PortaNum™", an assistive technology for visually impaired people. This every day tool uses images provided by a webcam or a digital camera and is based on image processing that leads to image focusing, contrast enhancement, contour detection and real time navigation. It contributes to better integration of visually impaired children since PortaNum helps reading better and interpreting pictures displayed on the screen. This freeware is available at <http://www.portanum.com/>

It is already used in several schools in France. Various feedbacks of users led us to modify the IHM in order to make it easier to use, more accessible for every one and more attractive. Furthermore, experiments with image processing gave satisfying results and many children daily use it as a technical tool to read the board. In addition, these positive feedbacks led to the development of a network solution. Only one camera is required unlike the precedent version which required at least one camera per student. It will dispatch the video stream to various computers through the network. thus enabling them to work on an image of high quality and perfectly suited to the board's dimensions.

To improve the usability of the software, we decided to make it the more accessible it can be following coherent guidelines and recommendation. This new version will then be upgraded by a network solution with one or more cameras. Our final objective is to install them as soon as possible in the schools partners of THALES.

KEYWORDS

Wireless, network, assistive technologies, webcam, reading aid, PortaNum, low vision, visually impaired people, classroom, software, UDP, multicast, image processing.

1. INTRODUCTION

Dans le cadre de mon cursus universitaire, au sein du DESS nouvelles technologies et handicaps sensoriels et physiques à l'université Paris VIII et du Master sciences et informatiques 2ème année, spécialité réseaux à l'université Paris VI (Jussieu), j'effectue un stage à l'intérieur du groupe THALES [A001]. Cette expérience de six mois est centrée sur le développement d'une solution réseau pour un logiciel de traitement d'images en temps réel.

Cette aide technique pour personnes malvoyantes a été développée par les ingénieurs du département « optique et optronique » du centre de recherche THALES TRT sous la direction du service de la Mission Insertion. Cette entité du groupe THALES a pour objectif d'être un centre de ressources et de conseils au service de tous les acteurs de l'entreprise confrontés au handicap.

Pour le choix de mon stage, plusieurs points me tenaient à cœur : je souhaitais travailler sur l'accessibilité et l'ergonomie logicielle ainsi que sur le développement réseau afin de parfaire mes connaissances, acquérir de l'expérience et enfin pouvoir me servir des compétences acquises lors de mon année de DESS. Le stage que propose THALES réunit toutes ces exigences. Il me permet, en effet, de développer un logiciel pouvant faciliter l'intégration d'un enfant malvoyant durant sa scolarité. Ce logiciel PortaNum™ dans une version réseau sans fil, associé à un ordinateur portable et à une caméra permet à un enfant de visualiser de manière adaptée le tableau ou n'importe quel support de cours.

Après une présentation du groupe THALES et des différents services dans lesquels j'ai eu l'occasion de travailler, seront développés dans ce rapport :

- ❑ La population des personnes malvoyantes
- ❑ L'amélioration de la version monoposte du logiciel, en terme d'accessibilité et d'architecture.
- ❑ L'étude et le développement étape par étape de la solution réseau
- ❑ Le fonctionnement et l'expérimentation de la version finale

Pour finir, je mentionnerai les bénéfices personnels que ce stage a pu m'apporter et je terminerai sur le bilan du stage.

2. RAPPEL DU SUJET DE STAGE

Le déroulement de ma mission au sein de l'entité Research & Technology sur le site de Palaiseau, et piloté par le service DRH France| Mission Insertion se décompose en trois parties principales : la prise de connaissance du projet, l'amélioration de la version monoposte du logiciel et enfin l'adaptation de cette version sur une solution réseau Clients/Serveur.

Le détail de ma mission est décrit ci-dessous avec l'ensemble des points devant être abordés durant mes 6 mois de stage, d'avril à septembre 2005 :

Approche logicielle du projet PortaNum™ :

- ❑ Etude sur la version actuelle (version expérimentale)
- ❑ Redéfinition du cahier des charges
- ❑ Documentation technique, document de référence
- ❑ Architecture logicielle
- ❑ Ergonomie logicielle
- ❑ Définition d'une politique de tests du logiciel
- ❑ Caractérisation des performances, taille mémoire, cadence de rafraîchissement ...

Amélioration de la version actuelle

- ❑ Etude d'une nouvelle architecture
- ❑ Réalisation de bibliothèques de traitements d'images
- ❑ Définition d'une liste d'action
- ❑ Architecture alternative de celle existante (multi-thread, exploitation de DirectX, ...)
- ❑ Analyse comparative
- ❑ Développement d'une proposition d'architecture

Etude d'une version réseau

- ❑ Analyse de l'existant et des besoins des utilisateurs
- ❑ Prise de connaissance des versions expérimentales
- ❑ Cahier des charges
- ❑ Analyse théorique
- ❑ d'une version réseau filaire
- ❑ d'une version réseau sans fil (WI-FI ..)
- ❑ des codes correcteurs d'erreur
- ❑ des solutions de type Stream vidéo / Images, unicast / broadcast
- ❑ Prototypage d'une solution
- ❑ Développement d'une version simplifiée
- ❑ Evaluation des performances (Qualité de service, débit effectif, ...)
- ❑ Exigences matérielles pour les préconisations
- ❑ Développement d'une version réseau (commercialisable)

L'objectif final de mon stage est le développement réseau du logiciel PortaNum™ ainsi que la modification de son Interface Homme Machine, en la rendant à la fois plus simple et plus accessible. Celle-ci devrait pouvoir être installée dès la rentrée 2005/2006 dans des classes accueillant des personnes mal voyantes.

3. THALES

3.1 Le groupe

THALES, anciennement Thomson-CSF, est l'un des grands groupes mondiaux de haute technologie fournissant des équipements électroniques et des systèmes innovants dans les domaines de la défense, l'aéronautique, les technologies de l'information et de la communication.

Les activités du groupe sont aujourd'hui organisées sur trois grands pôles :

- **L'Aéronautique** (10 000 personnes) : avionique et multimédia de cabine, Gestion du contrôle aérien, Simulation et entraînement.
- **L'Electronique de Défense** (33 000 personnes) : équipements électroniques, systèmes d'intégration, maîtrise d'œuvre et services pour forces armées : Systèmes aéroportés, Systèmes de Communication, Systèmes Optroniques, Systèmes Navals, Systèmes de Défense Aérienne.
- **Les Technologies de l'Information et des Services** (19 000 personnes) : systèmes d'informations et des services, Sécurisation des transactions électroniques.

3.2 THALES R&T

Regroupant aujourd'hui 500 personnes, TRT a été créé en juin 2001 par regroupement des capacités de recherche du groupe THALES.

TRT fournit au groupe et à ses divisions les nouvelles technologies, les connaissances et les compétences pour leurs évolutions.

Dans le cadre de mon stage, j'interviens dans le Département Optique et Optronique.

Les activités de ce Département s'articulent autour de 4 axes :

- le développement de nouvelles sources lasers,

- l'identification et le traitement optique du signal,
- les composants optiques, la visualisation et la sécurité
- La mise en place de projets nouveaux dans lesquels intervient l'optique

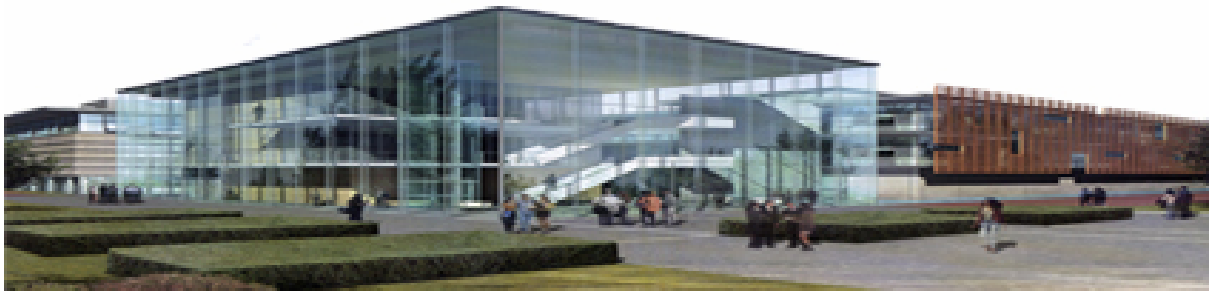


Figure 3-1 : Photo du site de TRT

3.3 Service MISSION INSERTION

Depuis 1992, THALES mène une action soutenue en faveur de l'insertion des personnes handicapées. Pour ce faire, au lieu de contribuer d'une façon obligatoire à un fonds : l'AGEFIPH (Association de Gestion des Fonds pour l'Insertion Professionnelle des Personnes Handicapées), le groupe a choisi de conclure un accord d'entreprise. Celui-ci vise l'implication de chacun à son niveau et avec ses compétences afin qu'il prenne en compte cette nouvelle dimension.

Dans cette optique, THALES a créé le pôle Mission Insertion dont le travail se découpe en 3 axes:

Axe 1 : La sensibilisation et l'information

- Informer et sensibiliser pour expliquer ce qu'est le handicap

Axe 2 : l'emploi

- Recruter les personnes handicapées dans les métiers de THALES et préparer les collaborateurs de demain,

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

- Accueillir des stagiaires et les former en créant des partenariats avec les grandes écoles, les universités, les centres de formation pour organiser des passerelles entre formation et entreprise,
- Maintenir la personne dans l'emploi en composant avec le handicap lorsqu'il devient une difficulté pour l'exercice d'une activité professionnelle,
- Accroître la sous-traitance avec le secteur protégé (ateliers protégés, centre d'aide par le travail).

Axe 3 : la technologie

- Ce pôle met en œuvre l'expertise technique et industrielle du Groupe au service de projets innovants visant à faciliter l'accès des personnes en situation de handicap au savoir et à l'emploi.

En ce qui concerne l'embauche, les résultats sont pour l'instant concluant puisque le taux d'emploi de personnes en situation de handicap est passé de 2.92% en 1992 à 5.85% en 2003.

Le troisième axe a permis le développement de plusieurs projets innovants :

- J'apprends la LSF : Ce projet terminé est une initiation sur cédérom à la pratique de la Langue des Signes Française, dédiée à l'entourage des personnes sourdes.
- J'apprends le LPC : Le principe du Langage Parlé Complété consiste à compléter les mouvements des lèvres par des gestes de la main (certains phonèmes étant difficiles à différencier sur les lèvres). Ce projet distribué sous forme de cédérom permet à toute personne d'apprendre cet encodage par l'apprentissage de plus de 800 mots de vocabulaire.
- Soundflyer : Ce projet technologique vise à permettre à des pilotes d'avions aveugles et malvoyants d'avoir une plus grande autonomie en restituant de manière sonore les informations de vol.
- Handi-aérosurveillance : L'objectif est la prévention et la vigilance des feux de forêts depuis un avion piloté par une personne handicapée moteur, via un système de surveillance infrarouge et de transmission au sol.
- Access Learning, plate-forme d'enseignement par Internet : L'objectif de ce projet, développé en partenariat avec l'institut de Garches, est de développer un dispositif d'enseignement à distance par Internet (e-learning). Il est destiné à

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

favoriser l'insertion professionnelle des personnes handicapées. L'innovation repose sur la capacité de suivre à distance un enseignement totalement individualisé.

- PortaNum : Logiciel d'aide à la vision de loin.

Ce dernier piloté par le service de la Mission Insertion fait l'objet de mon travail de stage ; nous allons le développer dans les parties suivantes.

3.4 Matériel mis à ma disposition

Durant toute la durée de mon stage, j'ai pu travailler sur un ordinateur de bureau {Ord1}¹ cadencé à 2.8GHz (1Go de mémoire vive) avec le logiciel de programmation Borland C++ Builder et avec une webcam et une caméra numérique. Les règles de sécurité du groupe THALES m'ont empêché de relier cet ordinateur avec tout matériel WIFI.

Pour les expérimentations sur la partie réseau de mon travail, j'ai eu accès au mois de juin à :

- un routeur WIFI 802.11b {R1}²
- un ordinateur portable {Ord 2} (sous Windows 98 Pentium III cadencé à 700 Mhz, 256 Mo de mémoire vive) possédant une carte PCMCIA 802.11b
- un ordinateur de bureau {Ord 3} (sous Windows 98 Pentium III cadencé à 700 Mhz, 256 Mo de mémoire vive)

Ensuite, au mois d'août j'ai pu travailler sur la norme 802.11g grâce à :

- un routeur WIFI 802.11g {R2}
- une carte PCMCIA 802.11g
- un ordinateur portable personnel {Ord 4} (sous Windows XP familial cadencé à 3Ghz avec 512 Mo de mémoire vive).
- une clé USB WIFI personnel

Au mois de septembre et dans le vue d'expérimentation de plus grand envergure :

¹ Référence permettant de définir cette ordinateur

² Notation pour les routeurs

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

- 3 PC portable avec des cartes PCMCIA 802.11g {Ord 5-6-7}
- 1 routeur 802.11g {R3}

4. UTILISATEURS FINAUX

Le logiciel d'aide à la vision de loin, PortaNum, cible essentiellement une population de personnes malvoyantes. Je vais donc commencer par faire une petite analyse sur cette population en donnant le cadre légal, les quelques chiffres, les différentes pathologies et les aides techniques.

4.1 Définition légale

L'Organisation Mondiale de la Santé définit le malvoyant comme une personne dont la déficience visuelle, après traitement et/ou correction est comprise entre 6/8 (0,3) ou dont le champ visuel est inférieur à dix degrés (10°) autour du point de fixation, mais qui utilise, ou est potentiellement capable d'utiliser sa vue pour planifier et/ou exécuter une tâche.

4.2 Chiffres sur la population déficient visuel

Il y a en France environ 1,6 millions de malvoyants, dont la répartition est la suivante :

- ❑ 207 000 personnes (12 % des déficients visuels) seraient aveugles ou malvoyants profonds, c'est à dire aveugles ou ayant une vision résiduelle limitée à la distinction de silhouettes, parmi lesquelles 61 000 seraient aveugles complets.
- ❑ 932 000 personnes (55 % des déficients visuels) seraient malvoyants moyens, avec une incapacité visuelle sévère en vision de loin (beaucoup de difficultés ou une incapacité totale à reconnaître un visage à quatre mètres) ou en vision de près (beaucoup de difficultés ou incapacité totale à lire, écrire ou dessiner).
- ❑ Enfin, un peu plus 560 000 personnes (33 % des déficients visuels) seraient malvoyants légers. La nature et l'origine des problèmes de santé déclarés par ces personnes ont conduit à les classer « malvoyants », mais elles n'ont pas déclaré d'incapacités visuelles sévères pour la vision de loin ou la vision de près. Pour cette raison, un grand nombre de ces malvoyants ne seraient probablement pas considérés comme déficients visuels selon les définitions légales ou les critères couramment utilisés dans la pratique médicale.

Le graphique ci-dessous montre la répartition de la population déficiente visuelle en fonction de l'âge. On comprend facilement que le premier facteur de malvoyance (moyen) est le vieillissement de la population.

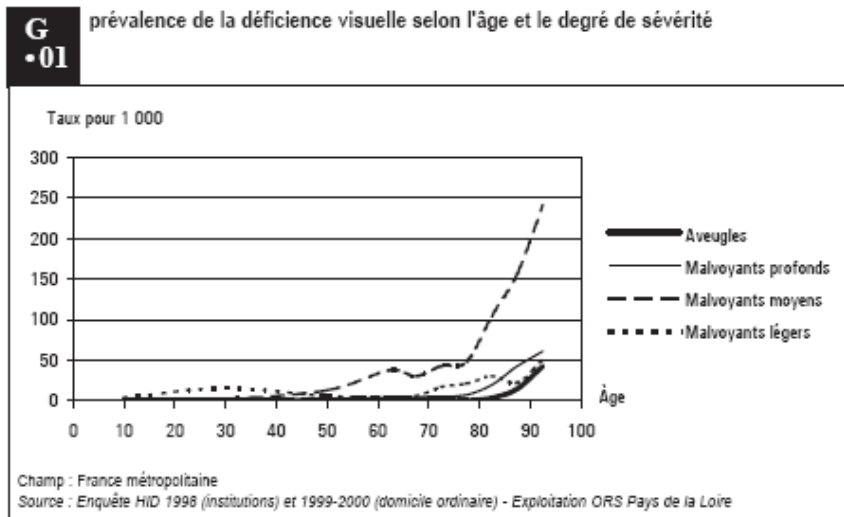


Figure 4-1 : prévalence de la déficience visuelle selon l'âge et le degré de sévérité

4.3 Types de déficiences

Les déficiences observées chez les mal voyants sont de plusieurs types. Citons entre autres:

□ Hémianopsie

Affaiblissement ou perte de la vue dans une moitié du champ visuel de l'un, ou plus souvent, des deux yeux.

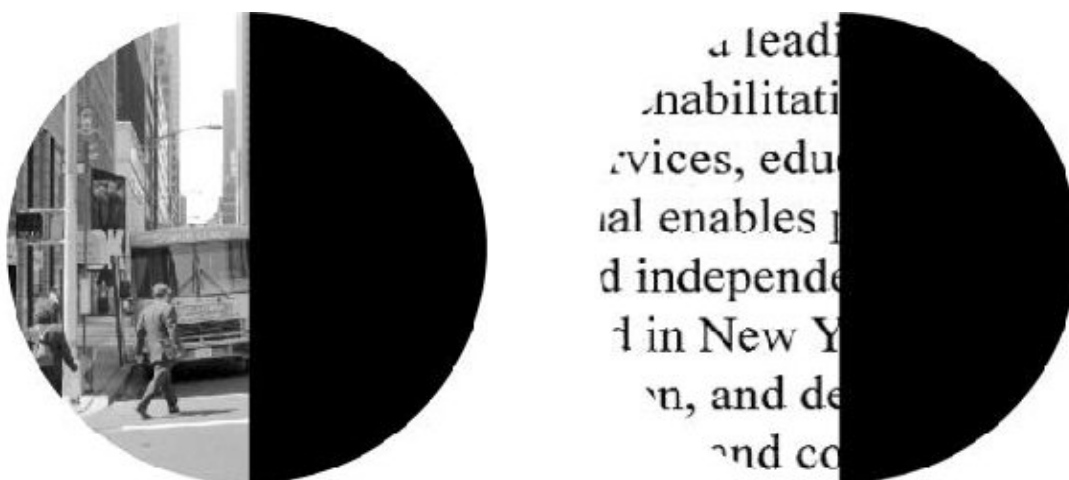


Figure 4-2 : Hémianopsie (Disparition de la moitié du champ visuel)

□ DMLA

La dégénérescence maculaire liée à l'âge est un véritable problème de santé publique, touchant plus d'un million de personnes en France. L'allongement de la durée de vie aboutit à une augmentation de cette maladie qui survient généralement à partir de 60 ans. Concrètement, la DMLA est la cause la plus fréquente de dégradations irréversibles de la vue des personnes âgées. La macula, située au pôle postérieur de l'oeil, est l'endroit où l'acuité visuelle est maximale. La rétine correspond au film d'une caméra; dans l'oeil, c'est elle qui capte les images à transmettre au cerveau. Sa partie la plus sensible, celle qui nous permet une vision de précision, qui nous permet de lire, d'écrire, d'enfiler une aiguille, bref d'exécuter tout travail minutieux, c'est la macula.

Chez une personne diabétique, lorsque la rétine est atteinte, on parle de rétinopathie diabétique.



Figure 4-3 : Exemple de vision altérée par le DMLA

- **Glaucome**

Le glaucome chronique est une maladie assez fréquente (1 à 2% de la population en France métropolitaine), touchant surtout les adultes après l'âge de 40 ans, mais pouvant également survenir chez l'enfant ou l'adulte jeune.

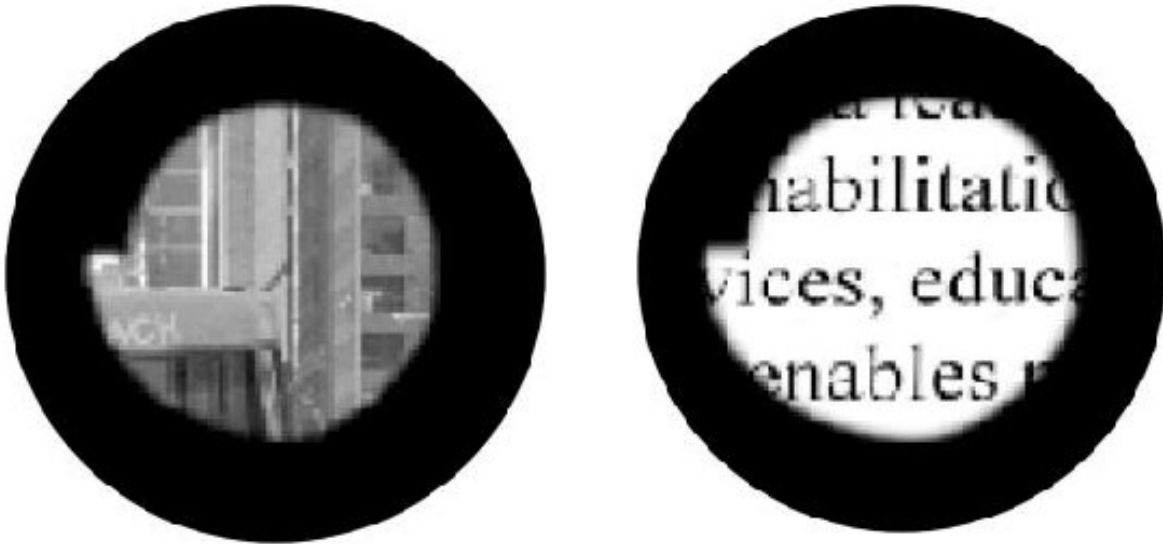


Figure 4-4 : Exemple de vision altérée par un Glaucome (destruction lente du nerf optique qui détériore insidieusement le champ visuel, avec menace de cécité en l'absence de traitement)

- **Rétinopathie diabétique**

Il s'agit d'une maladie grave de la vision, due à une atteinte de la rétine chez les personnes diabétiques.



Figure 4-5 : vision altérée par rétinopathie diabétique

4.4 Rapport aux aides techniques

Une personne malvoyante dispose d'un reste de vision dont elle voudra généralement profiter un maximum. Généralement, elle utilise un "logiciel d'agrandissement" (dépendant de l'anomalie visuelle, éventuellement combiné à un plus grand moniteur) qui agrandit l'image de l'écran, ce qui a pour conséquence de réduire la vue d'ensemble de l'écran. Via ce logiciel d'agrandissement, il est possible de transformer les couleurs de façon à obtenir un contraste plus puissant. Alors que le malvoyant trouve dans l'agrandissement l'adaptation la plus adéquate pour dactylographier un texte (par exemple dans un système de traitement de texte), le logiciel de "synthèse vocale" semble être une meilleure solution pour la lecture de documents (par exemple une page Internet). C'est la raison pour laquelle la plupart des malvoyants utilisent un logiciel d'agrandissement combiné à un logiciel de synthèse vocale, ce qui permet simultanément l'agrandissement et l'écoute des informations.

Ces techniques sont, malheureusement, loin d'être parfaites ; les principaux éléments qui perturbent les personnes malvoyantes sont:

- certaines couleurs ou contrastes de couleur
- certains motifs de fond
- certaines polices de caractères ou styles (p.ex. italique)
- certaines tailles ou variations de taille de caractères
- les clignotements et mouvements d'objet
- l'absence de perception globale

Comme nous venons de le voir, l'accès à l'informatique est un problème important du fait du grand nombre de personnes malvoyantes en France. Il est donc primordial d'apporter de nombreuses aides techniques mais aussi de les concevoir de façon parfaitement accessible.

5. PORTANUM™, VERSION MONOPOSTE

Maintenant que nous avons étudié plus en détails la population que PortaNum vise à aider, nous allons présenter le projet PortaNum™ puis décrire les améliorations apportées à la version monoposte au cours de mon stage. En effet, cette version, en constante évolution depuis 5 ans, avait besoin d'être reconstruite depuis le cahier des charges jusqu'à son architecture logicielle en passant par l'ergonomie de son interface.

5.1 Description du logiciel

PortaNum™ est un logiciel d'aide à la vision de loin et de près. Les captures d'images sont réalisées par deux outils différents selon l'éloignement. Pour les captures de loin, on utilise des technologies performantes telles que les caméras numériques qui possèdent des caractéristiques comme les résolutions d'image et les zooms optiques permettant de bonnes captures d'images. Quant à la vision de près, la capture se fait à l'aide d'une webcam qui est souvent plus maniable et plus simple d'utilisation.

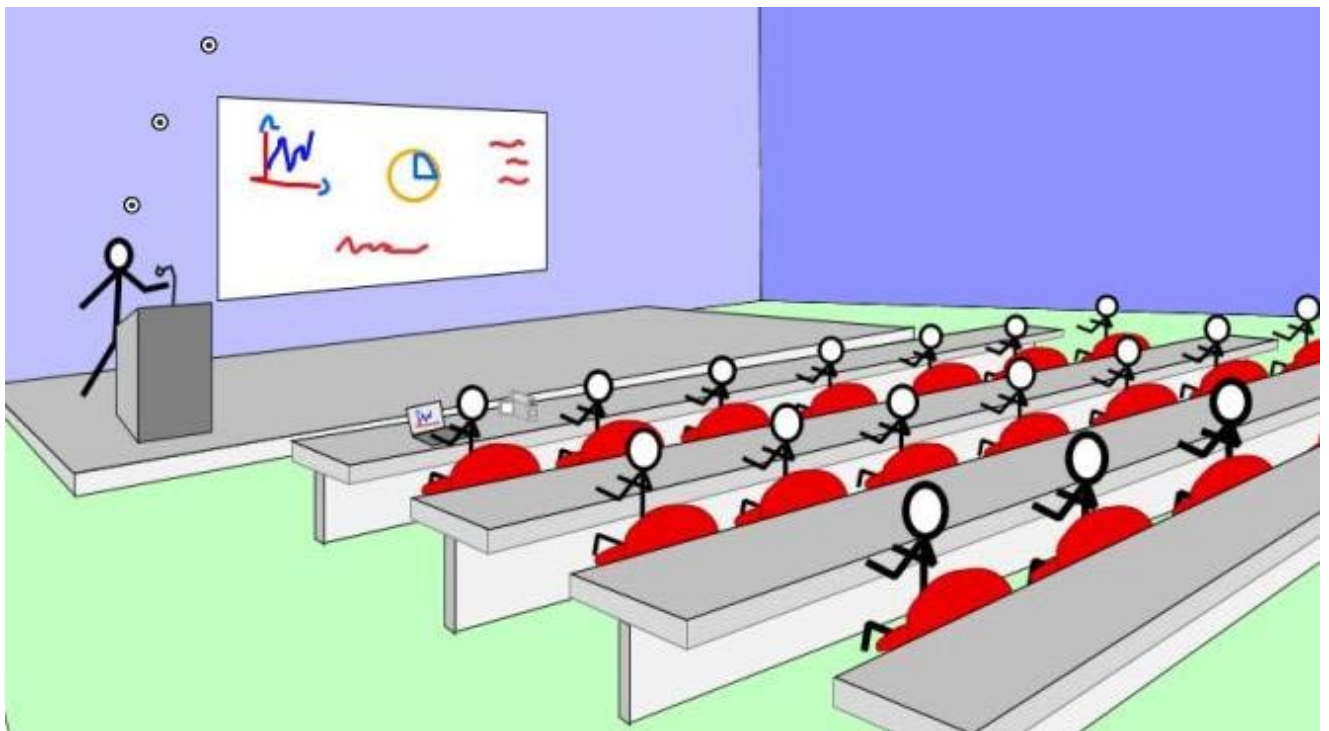


Figure 5-1 : Image tirée de la démonstration de PortaNum™. A l'aide d'un ordinateur portable et d'une caméra numérique, un élève malvoyant suit normalement un cours dans un

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

amphithéâtre. Grâce à PortaNum™, cet élève voit sur l'écran de son ordinateur des images améliorées et adaptées à sa vue

L'idée de base de ce logiciel est de prendre une image quelconque, c'est-à-dire à partir d'une webcam, d'une caméra numérique ou d'un disque dur, et de lui appliquer des fonctions de traitement d'images permettant ainsi de modifier les paramètres suivants, de manière interactive :

- Augmenter les contours
- Régler facilement les paramètres de luminosité, contraste, seuil
- Naviguer dans l'image (Zoom + mouvement directionnel grâce au glisser déposer)
- Sauvegarder des images brutes ou traitées.
- Prendre des notes sur un traitement de texte parallèlement à la visualisation
- Créer une vision en noir et blanc, fausses couleurs ou négatifs
- Faire ressortir du texte manuscrit ou imprimé à l'intérieur d'une image
- Gérer soi-même les caractéristiques de fort contraste et de fausses couleurs
- Passer facilement d'une caméra à une autre.

L'utilisateur peut gérer l'affichage des images en passant par l'un des 3 composants de l'Interface Homme Machine, soit la barre de menu, soit les raccourcis claviers, soit la boîte à outils. Ensuite, lorsque l'image est affichée, l'utilisateur a le choix de visualiser, d'enregistrer des images ou de prendre des notes.

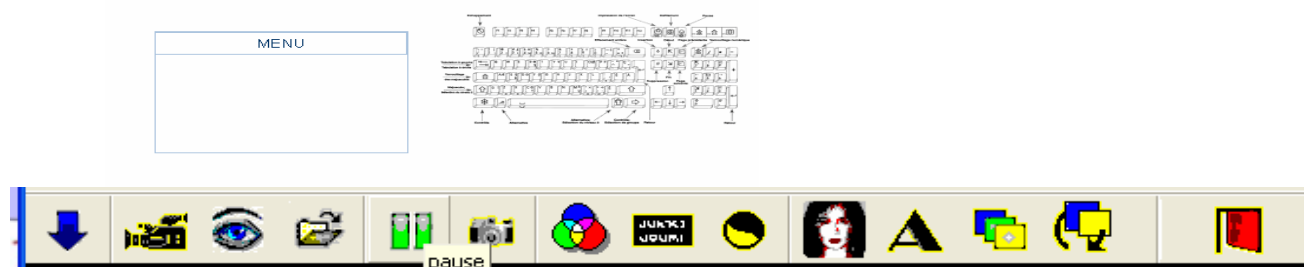


Figure 5-2 : Les 3 interfaces homme machine : la barre de menu, le clavier et la barre d'outil situé sous la barre de menu dans les versions 3.1, 3.2 et 3.3 de PortaNum.

PortaNum™ possède aussi de nombreuses fonctions de gestion d'images (enregistrement, suppression ...), permettant à un enfant malvoyant qui suit un cours de conserver les traces du tableau (qui lui sont vraiment utiles).

5.2 Accessibilité et ergonomie logicielle

Durant mon stage, nous avons décidé de retravailler toute l'interface Homme-Machine. Nous avons de ce fait commencé par remplacer la barre d'outils dont les images et la forme n'étaient pas des plus simples à manipuler, par une interface basée sur le principe d'une « télécommande » (nous la détaillerons ci-dessous). Finalement, l'interface Homme-Machine une fois modifiée sera composée de :

- La barre de menu,
- Les raccourcis clavier,
- La télécommande.

5.2.1 Simplification de la barre de menu

Comme je viens de le mentionner, mon travail a visé, dans un premier temps, l'amélioration du menu. Celui-ci comportait, en effet, plus de 48 items ce qui le rendait trop important et de ce fait complexe d'utilisation. De plus, il s'est avéré que certaines de ses fonctionnalités étaient peu ou pas employées par les utilisateurs. Nous avons donc opté pour une simplification de ce menu. Pour ce faire, nous avons décidé de regrouper toutes les actions les moins utilisées dans des fiches dédiées. Ces fiches décrivent les différentes options fonctionnelles permettant aux utilisateurs de personnaliser leur environnement de travail en fonction de leur besoin spécifique ainsi que de régler les traitements d'images et/ou de régler les préférences de visualisation. Ainsi, j'ai réduit le nombre d'actions à 29. Celles-ci sont directement atteignables par le menu en gardant, bien sûr, la possibilité pour l'utilisateur de modifier des options selon son souhait.

Vous trouverez la nouvelle interface de la barre de menu à la page suivante.

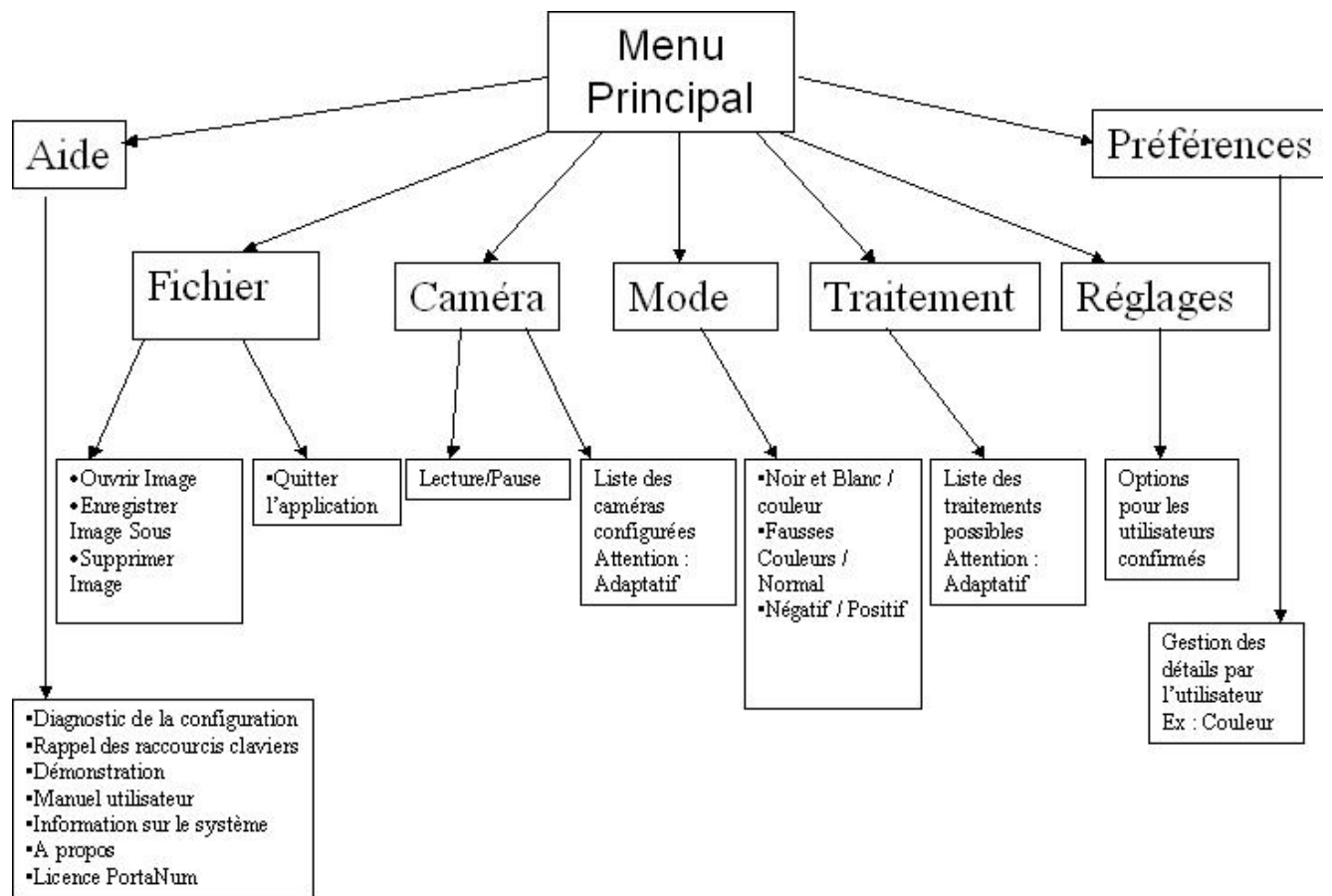


Figure 5-3 : Arborescence du Menu de la nouvelle version de PortaNum™

5.2.2 Raccourcis clavier

Les raccourcis clavier ont une part importante dans l'accessibilité ; en effet, ils permettent de pouvoir utiliser le logiciel sans utiliser la souris. Ceux ci sont très souvent utilisés par les personnes malvoyantes qui ont du mal à se diriger avec la souris. Ainsi, j'ai travaillé afin de donner à chaque action de l'interface une possibilité d'accéder en utilisant la souris. Pour chacune d'elle, j'ai vérifié à ne pas réutiliser des raccourcis clavier Windows qui permettent par exemple de naviguer entre les applications actives. J'ai vérifié à réutiliser des raccourcis clavier standard comme Ctrl + S pour enregistrer l'image.

Vous trouverez ci-après la page HTML contenant l'ensemble des raccourcis clavier utilisable avec PortaNum.

Tableau 5-1 : Les différents raccourcis clavier disponible dans PortaNum

Raccourcis-clavier pour utiliser PortaNum en mode Capture	
Raccourci	Action
Espace/RetourArrière	Passer au mode Image
Entrée	Lecture/Pause

Raccourcis-clavier pour utiliser PortaNum en mode Image	
Raccourci	Action
Barre Espace	Passer à l'image suivante dans le dossier
Ctrl + S	Sauvegarder l'image courante
Entrée/Ctrl - K	Passer au mode capture
Retour Arrière	Retourner à l'image précédente
Ctrl + X	Supprime l'image courante

Raccourcis-clavier généraux de PortaNum	
Raccourci	Action
L + flèches haut, bas et gauche	Modifier la luminosité
C + flèches haut, bas et gauche	Modifier le contraste
N + flèches directionnelles ou glisser déposer avec la souris	Passer en mode navigation
Z + flèches haut, bas et gauche	Agrandir une partie de l'image
P + flèches haut, bas et gauche	Modifier le paramètre du traitement en cours
Ctrl + S	Sauvegarder l'image courante
Ctrl + O	Ouvrir une image sous ...
Ctrl + C	Mode Couleur / Noir et Blanc
Ctrl + I	Mode négatif / Positif
Ctrl + F	Mode Fausses Couleurs / Normal
Ctrl + Z	Réinitialiser les traitements et les paramètres
Ctrl + N	Ouvrir le traitement de texte
F1	Afficher l'aide
F2	Diagnostic
F3	Information système
F4	A Propos
F5	Lancer la démonstration
F6	Passer au traitement suivant
F7	Passer au traitement précédent
F8	Affiche le menu propriétés du traitement en cours
F9	Affiche la récapitulatif des raccourcis clavier
F11	Plein Ecran
Traitement Automatique	1
Traitement Scène Sombre	2
Traitement Contraste de lumière	3
Traitement Texte imprimé	4
Traitement Texte manuscrit	5
Réglages Caméra	Alt + W
Préférence Images	Alt + I
Préférences Fausses Couleurs	Alt + G
Menu Expert	Alt + E
Préférence Affichage	Alt + A
Préférences Navigateur	Alt + N
Détecter Caméra	Alt + D
Choisir le traitement de texte	Alt + J

5.2.3 Développement d'une interface

Dans un deuxième temps, nous avons décidé de créer une nouvelle interface séparée de la fenêtre principale afin de faciliter l'utilisation des matériels de types contacteurs ou pointeurs et de remplacer la barre d'outils. Cette nouvelle fenêtre a été pensée pour faciliter l'accès de ce logiciel aux personnes malvoyantes ayant un trouble associé tel qu'un handicap moteur, intellectuel, etc. Ainsi, prenons par exemple l'application d'une fonction de type Zoom ou de navigation. Celle-ci requiert de nombreux allers et retours rendant de ce fait son accès difficile aux personnes ne pouvant pas utiliser les raccourcis clavier et/ou la souris. Il m'a semblé alors intéressant pour cette interface de sélectionner les actions les plus pertinentes afin de les disperser dans deux onglets différents. Le premier onglet regroupe les actions utiles, à savoir la navigation et le Zoom sur l'image, le second quant à lui regroupe les actions de réglages, permettant à l'utilisateur de modifier la luminosité, l'intensité ou le contraste, lorsque ce dernier entre dans une salle pour l'adapter à sa vue.

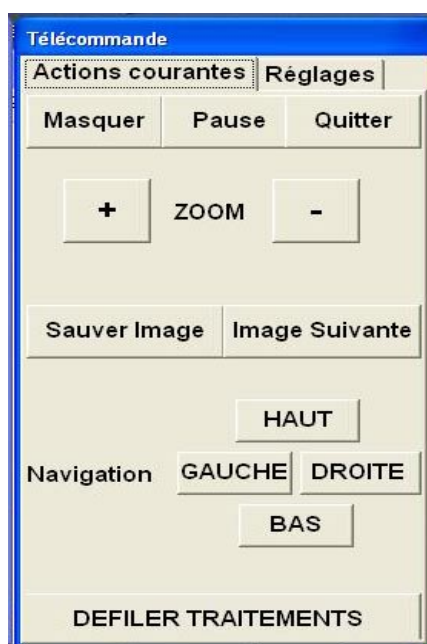


Figure 5-4 : Prototype de la télécommande

5.2.4 Retours visuels et sonores

Dans un troisième temps, nous avons décidé de supprimer le bloc-notes intégré dans l'ancienne version de PortaNum™. Ce dernier étant un traitement de texte assez rudimentaire, il n'était de ce fait pas utilisé dans la plupart des cas, les utilisateurs préférant les traitements de texte habituels tels que :

- WordPad fourni avec Windows,
- Microsoft Word (Pack Office),
- Jarte³ [A002] (un traitement de texte avec une IHM très accessible et paramétrable),
- WinPad⁴ 32 [A003] (un traitement de texte ayant en plus des possibilités de modifier les couleurs de fond).

Nous avons donc considéré, en nous appuyant sur l'avis des utilisateurs, qu'une application de traitement de texte devait posséder beaucoup plus de fonctionnalités que notre bloc-notes. De ce fait, dans la nouvelle interface, nous avons décidé d'ajouter un bouton, dans le menu fichier, qui permet de lancer un traitement de texte déjà personnalisé par l'utilisateur selon ses besoins. Ce choix du traitement de texte se fait dans une des fenêtres de réglages du logiciel.

Enfin une dernière particularité de cette interface Homme Machine est celle du retour d'informations : sonores et visuelles. En effet, pour une personne malvoyante, le recours à un effet sonore ou à un affichage de type gros caractères, est quelque chose de primordial. C'est pourquoi nous avons décidé de rajouter, à chaque action importante, ce type de feedback permettant à l'utilisateur de savoir que son action a bien été prise en compte.

- Un premier feedback est sonore, il émet un son léger à chaque action prise en compte par le système rendant l'interface à la fois plus conviviale et plus accessible.
- Le second retour d'information est visuel, il renvoie la valeur et le paramètre modifiés par l'utilisateur. Par exemple, la valeur de zoom , contraste, lumière, etc....

La capture ci-dessous montre l'exemple de ce retour visuel situé tout en bas de PortaNum™ pour ne pas gêner l'utilisateur.

³ Jarte n'est pas prévu particulièrement pour les personnes malvoyantes mais il offre beaucoup de facilité d'accès

⁴ WinPad 32 est l'équivalent accessible du WordPad de Windows.

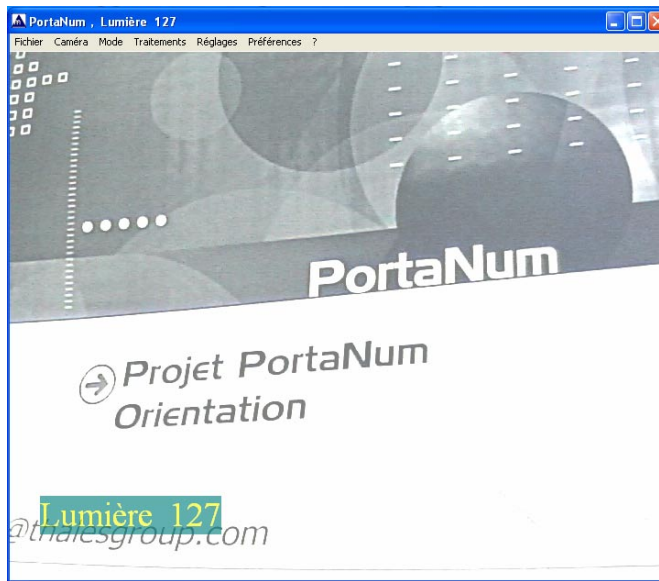


Figure 5-5 : Retour visuel sur le paramètre Zoom avec la valeur 1.4

- Le dernier retour d'information est aussi visuel. Cette visualisation permet sur une petite fenêtre de mieux gérer son déplacement dans l'image. Ainsi, la navigation est facilitée. Une autre utilité de cette fenêtre est de permettre la navigation dans l'image grâce au glisser/déposer. En effet, l'utilisateur peut bouger le rectangle symbolisant ce qu'il visualise. Ce mouvement aura comme répercussion le mouvement dans l'image.



Figure 5-6 : Retour visuel permettant la navigation dans l'image après avoir appliqué un zoom numérique. Ici, on visualise la partie inférieure gauche de l'image capturée. On pourrait déplacer le rectangle vert avec la souris afin de se déplacer dans l'image.

5.2.5 Couleurs, contrastes et police.

De nombreux troubles de la **malvoyance** entraînent une altération des couleurs ou des difficultés à différencier les couleurs peu contrastées. C'est pourquoi le logiciel PortaNum™ se doit de permettre à l'utilisateur de configurer lui-même toutes les parties de l'interface nécessitant des réglages d'accessibilités. Ainsi, j'ai ajouté pour le réglage des retours visuels, la possibilité de modifier les couleurs (de fond et de premier plan) ainsi que de tous les réglages de fond de texte. Les figures ci-dessous présentent les deux fenêtres de réglages permettant à l'utilisateur d'adapter la fenêtre à sa vue :

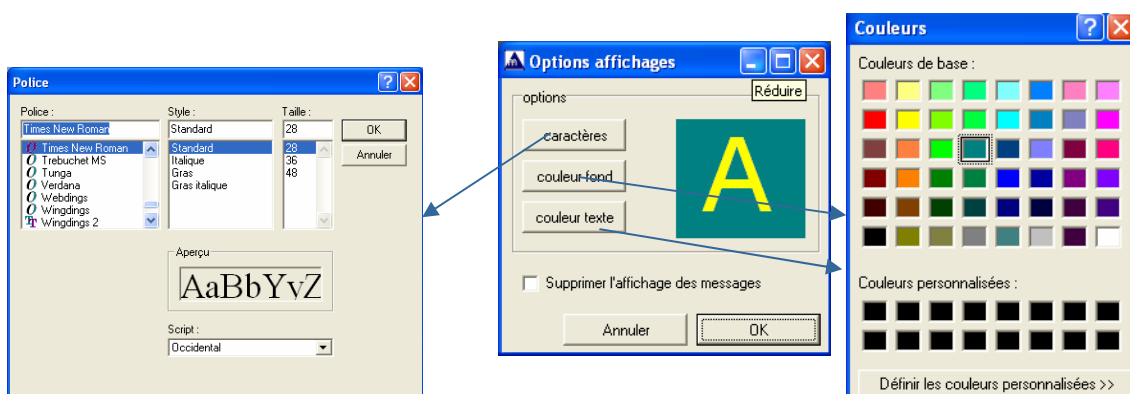


Figure 5-7 : Paramétrages du retour visuel des paramètres de contraste, lumière, zoom, ...

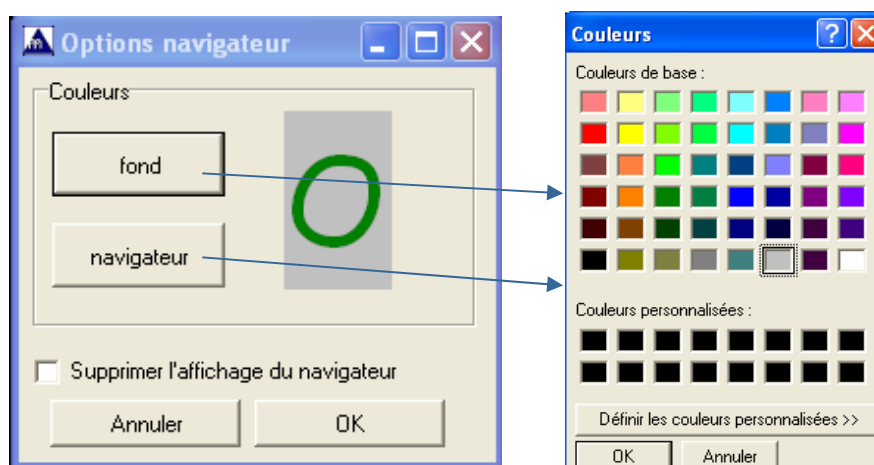


Figure 5-8 : Paramétrage du retour visuel de navigation. On peut configurer la couleur de fond et celle du premier plan.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Le même paramétrage de couleur est possible sur le mode fausses couleurs. Ce mode permettant de contraster au maximum les couleurs de départ de l'image capturée. Voici un exemple de visualisation en mode fausses couleurs :



Figure 5-9 : On a paramétré le mode fausses couleurs pour faire contraster en se basant sur un jeu de couleurs vert/jaune.

Par défaut, pour tous nos choix de couleurs, nous avons mis les couleurs vert et jaune (vert (R=0 ,V=128, B=128) pour le fond et jaune (R=255, V=255, B=0) pour le premier plan).

Comme décrit par le World Wide Web Consortium (W3C), la différence d'intensité entre la couleur de fond et la couleur de premier plan doit être supérieure à 125. La formule permettant de calculer la luminosité d'une couleur étant :

$$Brightness = \frac{(RedValue \times 299) + (GreenValue \times 587) + (BlueValue \times 114)}{1000}$$

On obtient donc :

- couleur de fond $BFond = 89.8$
- couleur de premier plan $BPPlan = 225.9$

La différence étant d'environ 136, le contraste est assez important pour ne pas gêner une personne malvoyante.

Le W3C indique aussi que la différence entre deux couleurs dont la formule est précisée ci-dessous doit être supérieure à 500.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

$$\text{ColourDiff} = \text{Max}(R1, R2) - \text{Min}(R1, R2) + \text{Max}(V1, V2) - \text{Min}(V1, V2) + \text{Max}(B1, B2) - \text{Min}(B1, B2)$$

On trouve pour nos couleurs par défaut :

$$\begin{aligned}\text{ColourDiff}(\text{Vert}, \text{Jaune}) &= 255 - 0 + 255 - 128 + 128 \\ &= 511\end{aligned}$$

Nos deux couleurs sont donc suffisamment différentes pour le contraste⁵.

Finalement, tout mon travail sur l'interface graphique de PortaNum a visé à la rendre la plus accessible possible.

Pour m'assurer d'une totale accessibilité, j'ai dû vérifier que l'interface prenait bien en compte les paramètres Windows. Ceux-ci peuvent modifier les couleurs des différentes fenêtres et du fond en les rendant mieux adaptés à une personne malvoyante. J'ai ajouté différents retours sonores et visuels pour la prise en compte des actions ainsi qu'une interface de type télécommande rendant l'accès à PortaNum plus facile lors de son utilisation avec des aides techniques de type contacteur.

Mon objectif final fut de remplir les recommandations les plus "pertinentes" de la CheckList d'IBM. Vous pourrez retrouver l'ensemble de ces conditions en Annexe.

5.3 Simplification de l'architecture des fonctions de traitement

Comme nous l'avons vu précédemment, le projet s'étant construit petit à petit avec le temps, nous avons du revoir l'architecture logicielle de PortaNum™. Celle-ci était plutôt simple, on trouvait :

- dans la fenêtre principale toutes les fonctions de traitement d'images, d'interfaçage avec le matériel, d'interaction avec l'utilisateur ...
- dans la classe de capture l'interfaçage avec la ou les caméras.

Pour mieux séparer les choses et rendre plus simple toute nouvelle amélioration ou toute nouvelle modification, nous avons décidé de décomposer cette architecture en 4 grandes parties :

- La fenêtre principale : qui gère les interactions entre les classes et les interfaces matérielles,

⁵ [Guideline 2.2](#) du Web Content Accessibility Guidelines 1.0

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

- L' IHM : qui est décomposée en éléments graphiques de Builder Borland C++, représentant les boîtes de dialogue de paramétrage.
- La classe de capture qui gère la cadence de capture ainsi que le format d'images obtenu par la caméra,
- Les Bibliothèques de traitement d'images, au format DLL, qui sont facilement intégrables par un système de plugin. Cette dernière amélioration a permis d'alléger le programme principal et aussi de le rendre évolutif.

Vous trouverez ci-dessous un schéma explicatif de l'architecture des fonctions.

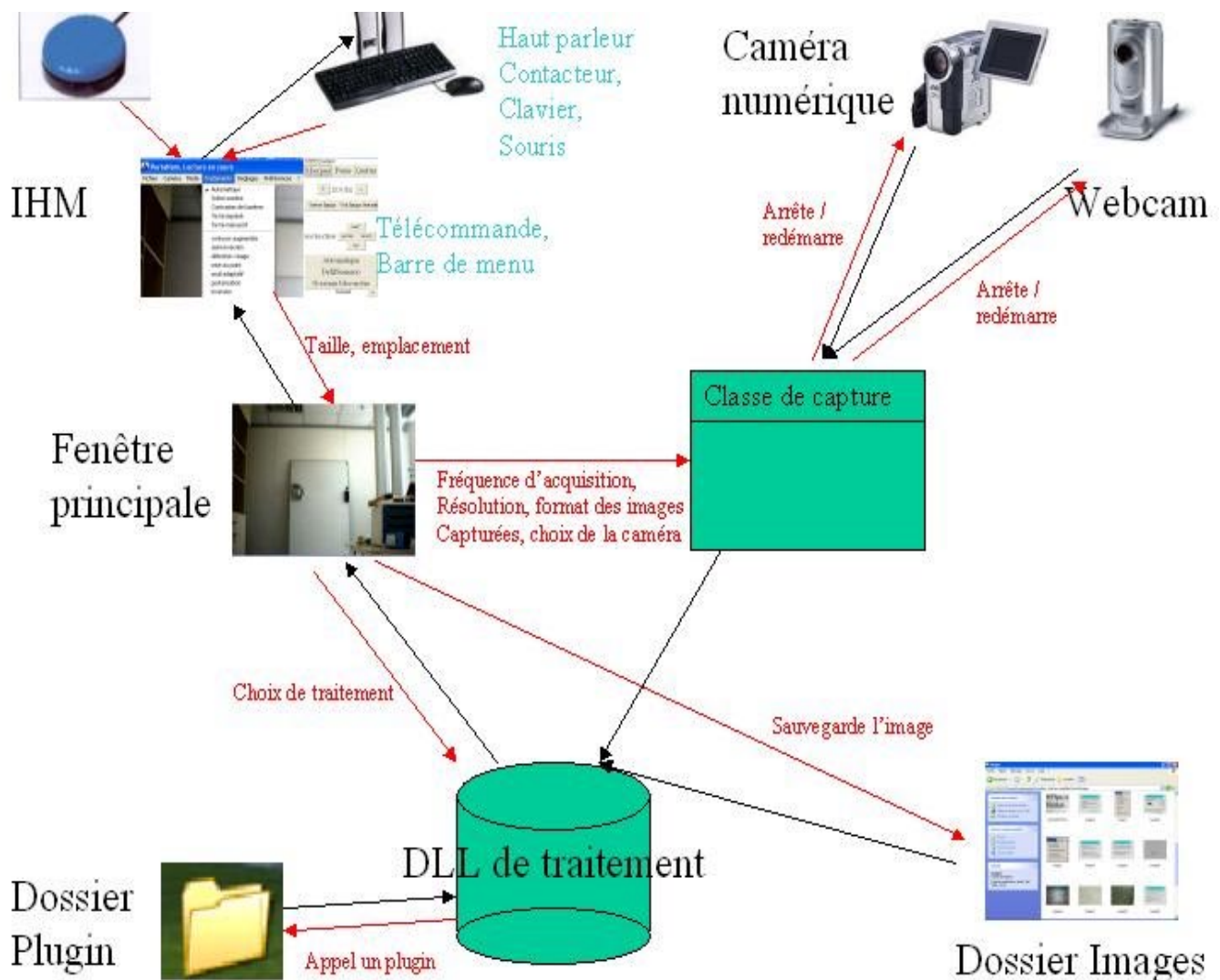


Figure 5-10 : Ce schéma présente l'architecture logicielle de PortaNum™.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Sur ce schéma, on remarque que la fenêtre principale est la partie fédératrice. En effet, elle peut :

- Gérer, en passant par la classe de capture, tous les paramètres de l'acquisition vidéo,
- choisir les traitements en appelant la bonne librairie ou le bon plugin,
- sauvegarder ou lire des images du disque dur,
- et interagir avec l'utilisateur grâce aux différents composants de l'interface.

La nouvelle version monoposte de PortaNum est une version plus facile à comprendre, mieux finalisée et mieux adaptée aux besoins de la population cible. En effet, elle n'est plus composée que de 5 traitements d'images qui s'adapteront à 90% des situations. Pour les 10% restants, elle donnera la possibilité à l'utilisateur de rajouter de nouveaux traitements dans PortaNum en téléchargeant simplement de nouveaux plugin sur Internet.

Nous allons décrire ci-dessous les 5 traitements de base⁶ puis nous décrirons le système de plugin :

Tableau 5-2 : Description des 5 fonctions de base directement intégrées à PortaNum

Fonction	Description
Automatique	<ul style="list-style-type: none"> • correspond à la majorité des utilisations • scène normale, nature des images variée, éclairément moyen, contraste d'éclairément moyen • l'utilisateur souhaite une image en couleur et demi-teintes, avec un contraste optimal
Scène sombre	<ul style="list-style-type: none"> • scène normale, nature des images variée, éclairément faible, contraste d'éclairément moyen • l'utilisateur souhaite une image en couleur et demi-teintes, avec un contraste optimal, y compris dans les zones les plus sombres
Contraste de lumière	<ul style="list-style-type: none"> • cas où l'éclairément des images est inhomogène (zones claires, partie de l'image dans l'ombre) • scène normale, nature des images variées, éclairément moyen, contraste d'éclairément fort • l'utilisateur souhaite une image en couleur et demi-teintes, avec un contraste optimal, dans les zones claires comme dans les zones sombres
Texte imprimé	<ul style="list-style-type: none"> • nature des images : texte dactylographié, éclairément quelconque, contraste d'éclairément quelconque • l'utilisateur souhaite une image binaire en luminance, avec éventuellement un certain rendu des couleurs saturées
Texte manuscrit	<ul style="list-style-type: none"> • nature des images : texte ou graphiques à la main, éclairément quelconque, contraste d'éclairément quelconque • l'utilisateur souhaite une image binaire en luminance, avec éventuellement un certain rendu des couleurs saturées. Les traits seront élargis pour une meilleure lisibilité

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Le système de plugin fonctionne par l'intermédiaire de bibliothèques dans lesquelles sont encapsulées des fonctions de traitements d'images. Chaque plugin doit avoir la même structure citée dans le tableau ci-dessous pour être facilement intégré et géré dans la fenêtre principale. Pour l'utilisateur, un plugin représente juste un nouveau traitement d'images (On allège ainsi la version de base du logiciel). Néanmoins, un utilisateur confirmé peut télécharger sur le site de nombreux plugins pour personnaliser sa palette de traitements.

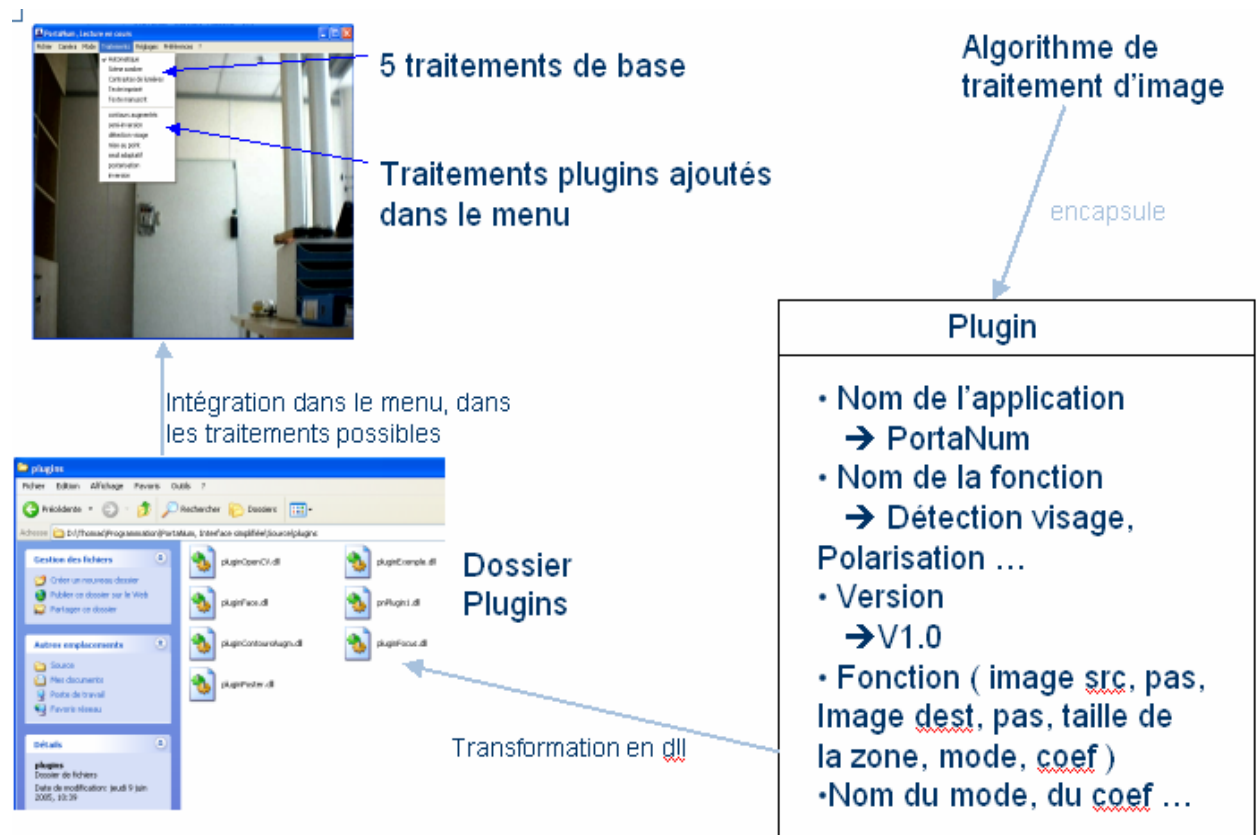


Figure 4-5-11 : Architecture du système de plugin. Les plugins sont ajoutés dans le menu principal lorsqu'ils sont placés dans le bon dossier. Ensuite, les fonctions sont paramétrables et directement applicables aux images capturées.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Après l'architecture en elle-même, nous avons trouvé intéressant de vérifier le système de communication entre les différentes classes. Celui-ci est pour l'instant un système de boucle qui fonctionne grâce à des flags indiquant quel traitement doit être appliqué durant ce tour de boucle.

C'est en vue de la création d'une version réseau qu'une évaluation d'un système de communication par message Windows a été réalisée (en opposition avec le système de fonctionnement par boucle des précédentes versions). Ce système permet à l'application de se remettre à capturer ou à traiter seulement lorsque cela s'avère nécessaire. De ce fait, le processeur est moins utilisé lorsque le rafraîchissement est inutile.

Après plusieurs expériences sur la charge CPU et les temps de traitement, nous avons remarqué que les deux systèmes étaient équivalents en performance. Nous avons donc remplacé l'ancien système par une communication par message donnant ainsi une plus grande liberté dans la gestion des appels entre les différentes classes, facilitant le cadencement. Ce cadencement est un point essentiel dans la version réseau puisqu'il permet une transmission parfaitement synchronisée.

6. PORTANUM™, VERSION RESEAU

La version réseau du logiciel est essentielle. En effet, elle permet de minimiser le nombre de caméras nécessaires lorsque PortaNum™ est utilisé par plusieurs personnes dans un même endroit. Une seule caméra parfaitement réglée va transmettre ainsi ces images pour qu'elles soient traitées par la suite selon les préférences des utilisateurs finaux.

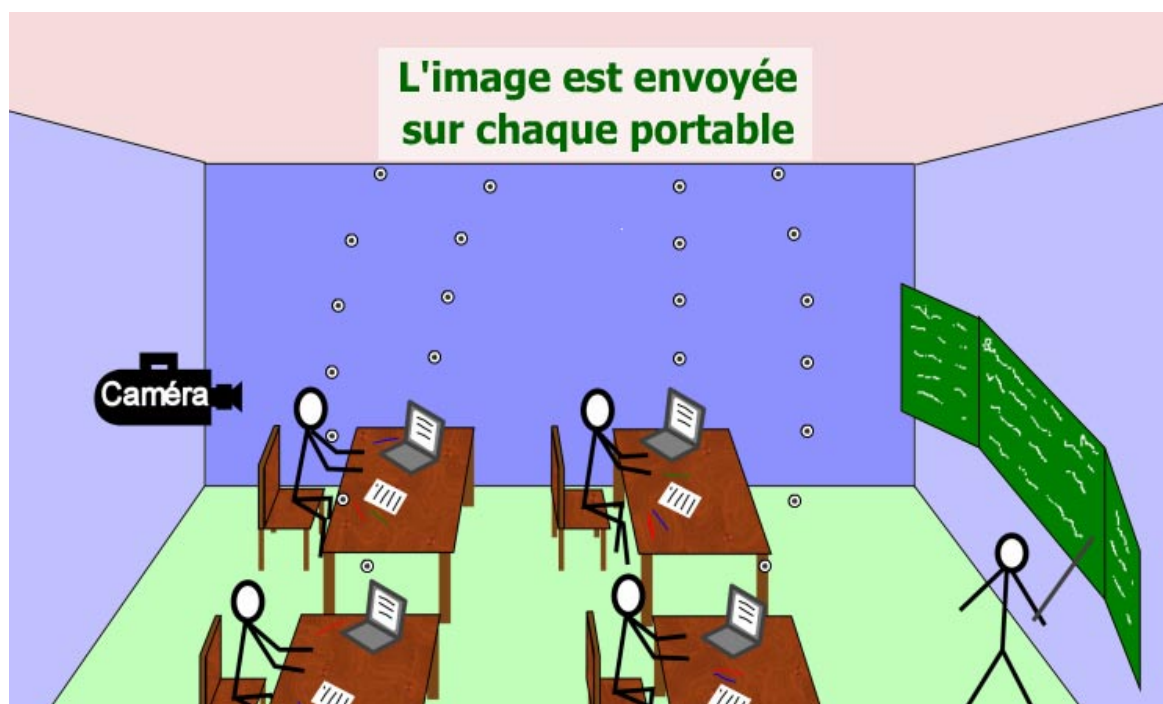


Figure 6-1 : Image tirée de la présentation de PortaNum™. On voit ici 4 élèves malvoyants suivant un cours sur leur ordinateur portable grâce aux images distribuées sur un réseau sans fil par la caméra numérique. Ils peuvent ensuite appliquer les fonctions de traitement d'images adapté à leur vue.

Nous présenterons dans le paragraphe à venir, quelles sont les principales demandes pour une version réseau. Nous détaillerons, par la suite, les différentes étapes du développement, et nous finirons par le déploiement de cette solution.

6.1 Analyse des besoins

Faire du logiciel PortaNum™ une application réseau consiste simplement à transmettre, à partir d'un serveur, des images de tailles variables vers des clients au sein d'un

réseau local. Ces images peuvent provenir d'une webcam, d'une caméra numérique ou d'images enregistrées sur le disque dur.

Dans le but de remplir ces objectifs, nous allons déterminer les différents besoins au niveau du réseau ainsi qu'au niveau de notre application.

6.1.1 Vue globale

Les différentes possibilités s'offrant à nous dans la proposition d'un réseau se décomposent selon les différentes couches du modèle OSI représenté ci-dessous :

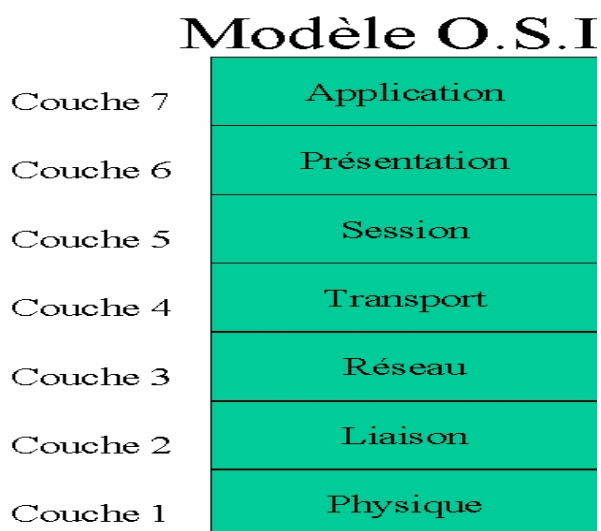


Figure 6-2 : Schéma représentant les couches du modèle OSI (Open System Interconnection).

Nous allons vous présenter les différents besoins de notre application de la couche physique à la couche application du modèle OSI.

Pour les couches Physique et Liaison de données, nous pouvons déjà émettre une question sur le type de matériels et de normes à utiliser dans les différents sous réseaux (défini par l'IEEE⁷ [A004] et plus particulièrement le comité 802) :

- ❑ Réseau filaire : token bus 802.4, token ring 802.5, Ethernet 802.3 ...
- ❑ Réseau sans fil : WI-FI⁸ [A005] 802.11, WIMAX⁹ 802.16...

⁷ L'IEEE s'occupe de la standardisation de systèmes électroniques et informatiques afin d'offrir une compatibilité à des produits de différents constructeurs. Le comité 802 s'occupe uniquement de la standardisation des réseaux locaux.

⁸ WI-FI (Wireless Fidelity) est une norme d'interopérabilité attribuée par la WI-FI Alliance aux produits issus des standards 802.11 certifiés et testés par cette organisme.

⁹ WiMax pour WorlWide Interoperability for Microwave Access

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Notre logiciel doit pouvoir s'adapter facilement aux différentes normes citées ci-dessous. En effet, la version réseau de PortaNum™ pourrait encourager une école ou une entreprise, ne possédant pas d'infrastructure réseau, à s'équiper. Compte tenu de la popularité des standards, il nous paraît essentiel de faire fonctionner parfaitement n'importe quel réseau filaire ou sans fil (du point de vue des erreurs de transmissions ou des délais 6.1.2.1 et 6.1.2.2).

Tableau 6-1 : Les principales normes IEEE 802.x. Vous trouverez les spécifications précises sur le site de l'IEEE

Norme IEEE	Titre et commentaires
802	Normes pour les réseaux locaux (LAN) et métropolitains (MAN)
802.1	Gestion et pontage des LAN et des MAN (y compris le protocole Spanning Tree)
802.2	Contrôle de lien logique
802.3	Méthode d'accès CSMA/CD (détection de porteuse avec accès multiple)
802.3u	Fast Ethernet
802.3z	Gigabit Ethernet
802.4	Méthode d'accès à passage de jeton sur un bus
802.5	Méthode d'accès Token Ring
802.6	Méthode d'accès DQDB (double bus de file d'attente distribuée) pour les WAN

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

802.7	LAN à large bande
802.8	LAN et MAN à fibre optique
802.9	Intégration de services (interconnexion de réseaux entre sous-réseaux)
802.10	Sécurité des LAN/MAN
802.11	LAN sans fil (une bande de base IR et deux signaux hyperfréquences à l'intérieur de la bande de 2 400 à 2 500 MHz)
802.12	LAN à haut débit (signaux à 100 Mbits/s, utilisant le mécanisme de demande de priorité)
802.14	Méthode d'accès de télévision par câble

En ce qui concerne la couche réseau, le passage par le protocole IP permet l'interopérabilité effective entre les différents sous réseaux. En effet, nous pouvons imaginer qu'un serveur transmette des images à travers une connexion sans fil et une connexion câblée. Une autre possibilité, au niveau de cette couche, est la création d'un groupe de diffusion possédant une information commune : l'adresse IP du groupe.

Pour la couche transport, il est important de déterminer le meilleur protocole entre UDP et TCP en se basant sur le taux d'erreur, le nombre de clients connectés et le matériel physique sous jacent.

Finalement, du point de vue de notre application, il est nécessaire de choisir entre différents modes de transmissions pour notre architecture réseau (broadcast, unicast, multicast), différentes possibilités de connexion, de synchronisation, de recouvrement d'erreurs, de compression de données ou de cryptage.

L'analyse des besoins à travers les expériences passées ont permis de faire ressortir plusieurs points à respecter lors du développement de notre application :

- ❑ Une assez bonne qualité d'image.
- ❑ Un nombre d'images par seconde supérieur à 2 ou 3.
 - ➔ Il est, en effet, souvent plus pratique de travailler sur une image quasi fixe lorsque l'utilisateur veut recopier le tableau.
- ❑ Un délai inférieur à quelques secondes.
 - ➔ Afin de ne pas avoir un trop grand décalage entre l'explication du professeur et l'image capturée.
- ❑ Le temps de démarrage de l'application peut durer quelques dizaines de secondes.
- ❑ Les données ne sont pas confidentielles.

D'un point de vue réseau, on peut donc avoir :

- ❑ Une limite de taux de paquets perdus assez importante
- ❑ Un temps de connexion et de synchronisation lent
- ❑ Un délai de bout en bout assez important
- ❑ Un protocole de transport sans retransmission
- ❑ Un taux d'erreurs faible ou un bon code correcteur d'erreurs
- ❑ Un cryptage basique, voire même aucun cryptage.

6.1.2 Choix du protocole de transport : TCP ou UDP

6.1.2.1 Débit

Un des points important, dans le choix du réseau, est bien sûr le **débit utile** (différent du **débit théorique** surtout avec une norme 802.11). Ce débit permet de savoir le nombre d'utilisateurs de PortaNum™ que nous pouvons accepter simultanément dans le cas de sessions unicast mais également de connaître le nombre d'images par seconde que nous pouvons espérer transmettre.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Vous trouverez ci-dessous un tableau récapitulatif des débits sur un LAN¹⁰.

Tableau 6-2 : Débit utile des standards les plus populaires sur un LAN.

Standard	Debit utile (Mbit/s)
Ethernet 10	8.08
Ethernet 100	90.06
802.11 (2 Mbit/s)	1.6
802.11b (11 Mbit/s)	6.56
802.11a (54 Mbit/s)	20.6
802.11g (54 Mbit/s)	22.6

Pour commencer, nous allons examiner en détail les conséquences du débit, du taux de compression et de la résolution de l'image sur le nombre d'images par seconde.

Vous trouverez, ci-dessous, un tableau récapitulatif des temps de propagation selon différents formats de compression.

Tableau 6-3 : Temps de propagation (en seconde) en fonction du format de l'image et du débit utile.

FORMAT / DEBIT utile (Mbit/s)		5	10	11	27.5	54
JPEG (50 ko et 70ko)	640x480	0.080	0.04	0.036	0.014	0.007
	720x576	0.112	0.056	0.051	0.020	0.010
JPEG 2000	640x480	0.040	0.02	0.018	0.07	0.0035
	720x576	0.056	0.028	0.025	0.01	0.005
SANS	640x480	1.476	0.738	0.671	0.27	0.137
	720x576	1.990	0.995	0.904	0.362	0.184

¹⁰ LAN : local area network, interconnexion d'ordinateur sur une circonférence d'un centaine de mètres. Au-delà, on considère que le réseau fait partie d'une autre catégorie de réseau appelé MAN (*metropolitan area network*), pour laquelle les supports de transmission sont plus adaptés aux grandes distances.

Dans le cas d'une image sans compression, un pixel est codé sur 24 bits et les tailles d'images sont donc :

- $24 \times 640 \times 480 = 7.38$ Mbits
- $24 \times 720 \times 576 = 9.95$ Mbits

Ces tailles sont beaucoup trop importantes pour vouloir les envoyer sans compression à travers un réseau.

Pour déduire le temps de propagation, on a utilisé la formule suivante :

$$\square \text{ Temps} = \frac{\text{nbBits}}{\text{Débit}}$$

Avec ce temps de propagation, on obtient facilement le nombre d'images par seconde que l'on peut espérer obtenir avec plusieurs débits de lien.

Tableau 6-4 : Nombre d'images par seconde en fonction du débit du lien avec une compression JPEG sur des images de résolution 640 x480.

DEBIT (Mbit/s)	Nombre d'images par seconde possible
5	12.5
10	25
11	28
27.5	71.42
54	143

On remarque que pour un débit de 10Mbit/s (facilement atteignable sur un réseau Wi-Fi 802.11g), le nombre d'images par seconde est équivalent à celui du cinéma (environ 24 images par seconde).

En conclusion, avec une bonne réception, il est facile d'atteindre une bonne fluidité d'affichage. Cependant, nous allons voir dans la suite qu'un réseau sans fil possède un taux d'erreur beaucoup plus important qu'un réseau filaire.

6.1.2.2 PER : Packet Error Rate

Dans une connexion basée sur le protocole Wi-Fi, il est courant de constater un taux d'erreurs de 8%. (J'ai trouvé cette information sur les caractéristiques constructeur de différents récepteurs Wi-Fi).

Ce taux d'erreurs nous amène à penser que garder une fluidité dans l'affichage, nécessite de pouvoir se reposer sur une retransmission des paquets perdus ou sur une correction d'erreurs au niveau récepteur (même si cela diminue la rapidité de transfert et ainsi le nombre d'images par seconde). En effet, il vaut mieux conserver une certaine fluidité même si on perd des paquets.

Finalement, malgré son manque de fiabilité, il nous est apparu préférable d'opter pour une application de type multicast basée sur le protocole UDP. En effet, le protocole TCP nécessiterait une connexion par client ainsi qu'une retransmission par paquet perdu.

Notre choix se base également sur l'idée que le réseau local dédié à notre application ne subira pas de congestion. Les seuls problèmes pouvant provenir du réseau sont des erreurs de transmission. Celles-ci peuvent être facilement corrigées en décomposant une image en plusieurs paquets de petites tailles puis en y ajoutant de la redondance. En effet, une erreur de bits sur un petit paquet donnera une perte peu conséquente. Ces paquets peuvent alors être reconstitués en fin de transmission plus facilement grâce à un calcul effectué entre les bits bien transmis et la redondance.

Après l'évaluation de différents codes FEC (Forward Error Correction : code correcteur d'erreurs), notre choix s'est rapidement porté sur le code « Reed-Solomon ». En effet, celui-ci permet une robustesse au niveau des pertes de paquets d'environ 10% avec des paramètres de redondance bien choisis. Le principe de ce codage est de rajouter de la redondance pour chaque mot de code ¹¹de taille fixe. L'entrelacement des symboles des mots de code permet de reconstruire une image après une perte de paquets de type « burst » (i.e plusieurs paquets perdus à la suite) durant la traversée du réseau.. En effet, le code Reed Solomon gère mieux les pertes de symboles bien réparties dans les différents mots de code.

Voici un petit exemple du fonctionnement de l'entrelacement et du code Reed Solomon [A006 A007] :

¹¹ Un mot de code est une partie des données servant de paramètre d'entrée pour le calcul de la redondance. On nomme en général « symbole » la plus petite unité composant le mot de code.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Imaginons que nos paquets nommés TS fassent 3 octets de long et que l'on transmette le paquet suivant: **03 10 15** auquel on ajoute deux octets de redondance. Le premier est la somme de nos trois données soit **28**, le deuxième est la somme pondérée des 3 octets. Chaque octet est multiplié par son rang : $3 \times 1 + 10 \times 2 + 15 \times 3$ soit **68**. Notre paquet à la sortie du codeur devient donc : **03 10 15 28 68** Après transmission et perturbation, le récepteur reçoit le paquet : **03 12 15 28 68** On refait la somme simple $03 + 12 + 15 = 30$ et la somme pondérée $3 \times 1 + 12 \times 2 + 15 \times 3 = 72$. La différence des sommes simples ($28 - 30$) nous donne la valeur de l'erreur et la différence des sommes pondérées divisée par l'erreur est égale au rang de l'erreur $(72 - 68) / 2 = 2$. On peut corriger notre paquet.

Pour mieux comprendre l'entrelacement, imaginons la suite de paquet de 5 octets suivante:

niche
chien
poule
tasse
fours
balles
video
audio
petit
porte

Chaque paquet est mis en mémoire horizontalement et relu verticalement. Ainsi si on relit les paquets depuis le e de niche, cela donne les paquets suivants:

eeuaf nlsob esuav erlia sldup eedep soito

A la suite de la transmission, on reçoit la séquence suivante affectée d'erreur:

eeuaf nlsob esuav **lia s***p eedep soito

Après désentrelacement, on retrouve les mots suivants :

e ---- tass* fou*s bal*es vi*eo ----

On voit nettement que l'entrelacement a permis de répartir les erreurs sur plusieurs paquets. Seul un octet par paquet est affecté et le code Reed Solomon peut facilement corriger ces erreurs. Sans entrelacement, ce sont les mots complets qui auraient été affectés sans espoir de pouvoir les récupérer.

6.1.3 Evaluation des composants Indy (flux multicast UDP)

Dans la continuité du projet PortaNum™ et des versions précédentes, j'ai réutilisé les composants réseau du logiciel Borland Builder C++ : Indy [A008] v9.00.14. Il est à noter que ces composants, utilisant la librairie « Winsock » de Windows, ne s'occupent essentiellement que de la couche application du modèle OSI.

Dans un premier temps, j'ai donc testé ces composants sur un flux vidéo multicast UDP composé de deux images fixes envoyées en boucle. Pour ces expériences, nous avons souhaité déterminer les performances au niveau du débit, des pertes de paquets (à quel endroit) et du nombre d'images par seconde, en fonction des différentes valeurs suivantes :

- délai inter-frames
- délai inter-images
- code correcteur d'erreurs
- taille des images
- taille des paquets de données au niveau application
- distance entre émetteur/ récepteur
- objets pouvant provoquer des interférences
- entrelacement des paquets.
- réception d'un paquet multiThreadé

Notre logiciel de test se nomme UDPTools, quelques captures d'écrans se trouvent en Annexe, vous permettant de mieux imaginer son fonctionnement.

Dans un deuxième temps, j'ai commencé la phase d'expérimentation faite sur deux ordinateurs {Ord1} et {Ord2} connectés par un routeur Wi-Fi {R1} selon le schéma suivant :

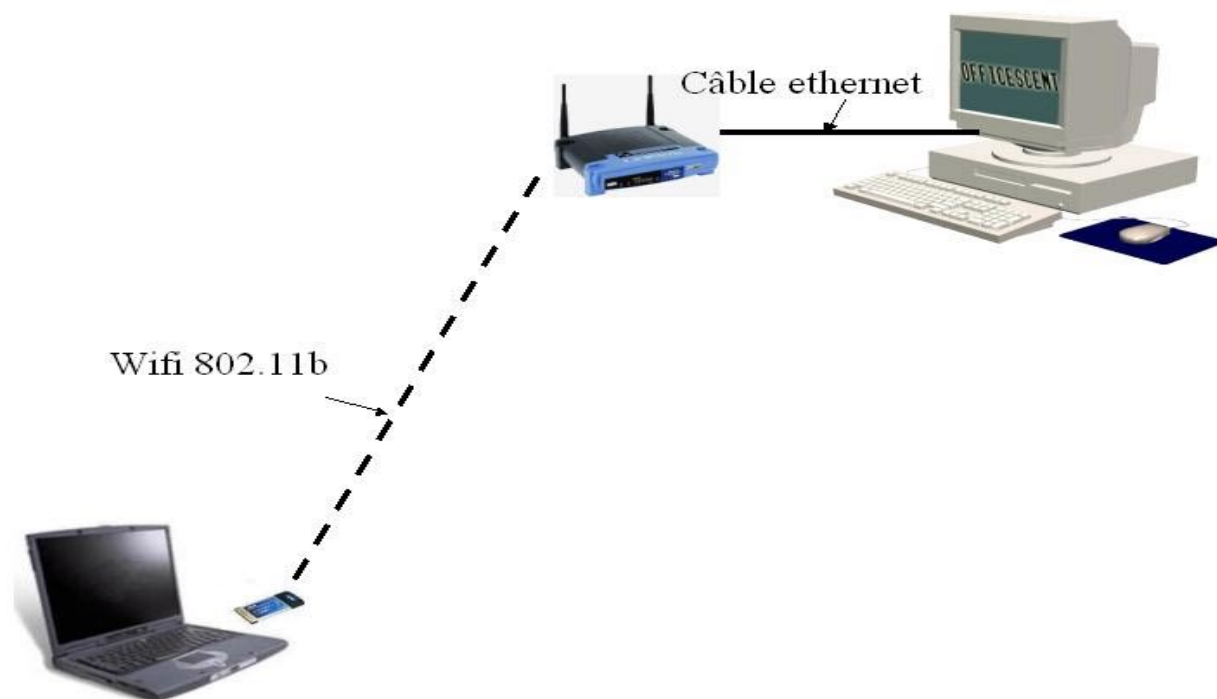


Figure 5-6-3 : Schéma de la connexion avec un débit moyen sur la liaison de 286Ko/s (mesuré par l'outil TestLab2003)

Expérience 1 : Réception multithread

Pour débiter, j'ai trouvé intéressant de savoir si une réception lancée dans un thread permettrait de gagner du temps par rapport à une réception séquentielle.

Avec différentes tailles d'images et plusieurs délais inter-paquets imposés au niveau de l'émetteur, les résultats sur 100 paquets émis indiquent qu'une réception avec un ou plusieurs thread donne un taux d'erreur nul. Ce paramètre des composants INDY ne semble pas critique pour notre application.

Expérience 2 : Rafraîchissement d'images en fonction de la taille des messages UDP

Le deuxième point qu'il m'a paru important de regarder est le nombre d'octets à émettre par paquets UDP. En effet, une grosse décomposition permet de n'envoyer qu'un seul entête, tandis que l'envoi de plusieurs paquets de petites tailles permet, en cas d'erreurs, de ne perdre que des morceaux de l'image facilement récupérables grâce à un code correcteur d'erreur.

Il est à noter que les composants Indy émulant le protocole UDP ne renvoie pas à l'application les erreurs de bits mais supprime simplement les messages UDP.

Tableau 6-5 : Rafraîchissement des images en fonction des tailles des messages UDP.

Taille des messages UDP	300	13 000	29 000	40 000	Image complète
Nombre d'images par seconde	2.56	3	3.57	3.57	4.35

Ce test indique que plus le paquet est grand, plus l'envoi d'une image est rapide. Une des raisons est l'augmentation du nombre d'entête (un entête par paquet), l'autre raison est le temps de passage, pour chaque paquet, dans le bout de programme émulant les différentes couches du modèle OSI.

Prenons une taille de 1 300 octets par paquets, on est en dessous de la valeur du MTU (1500 dans notre expérience, caractéristique d'une connexion Ethernet, en Wi-Fi le MSDU¹² est de 2312). On peut donc supposer qu'il n'y a pas de découpage de ces paquets lors de la traversée du réseau. Les images de notre expérience (47Ko) nous donnent une découpe en 37 paquets. Avec l'encapsulation des entêtes ci-dessous :

- ❑ 34 octets → Trame WI-FI
- ❑ 20 octets → Paquet IP
- ❑ 8 octets → Message UDP
- ❑ 2 octets → Application de test (pour connaître le numéro d'image et le numéro de paquets dans l'image) .

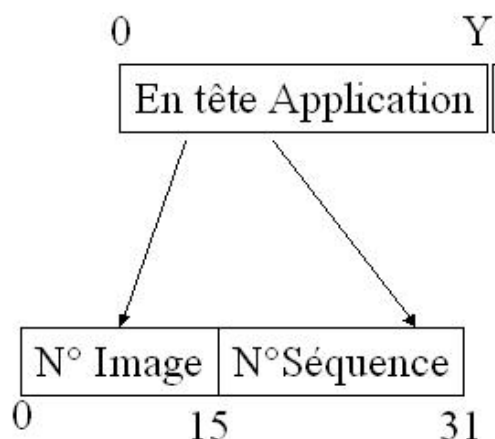


Figure 5-6-4 : Descriptif de l'entête de notre application de test. Le numéro d'image permettant de déterminer le passage d'une image à une autre tandis que le numéro de séquence permet de vérifier les pertes de paquets.

¹² MAC Service Data Unit

Message UDP

port source (16 bits)	port destination (16 bits)
longueur (16 bits)	checksum (16 bits)
données	

Figure 6-5 : Format de l'entête UDP sur 8 octets. Le checksum et la longueur permettent de détecter des erreurs de bits. Un message erroné sera supprimé par le composant INDY.

On trouve par image :

$$\rightarrow 37 \times 64 = 2368 \text{ octets d'entête.}$$

Finalement, nous obtenons donc avec cette découpe une perte de :

$$\frac{2368}{46920} \times 100 = 5.05\% \text{ (taille des données inutiles sur les données utiles).}$$

Nous pouvons alors comprendre pourquoi le débit chute avec des tailles de paquets plus petites. En effet, Un message UDP de 46 Ko sera seulement découpé au niveau de la couche liaison de donnée, économisant ainsi l'entête IP, UDP et Application. La perte du débit utile est pour ce format de message de : $\frac{32 \times 34}{46920} \times 100 = 2.31\%$. On voit donc que l'on perd moitié moins d'octets utile en envoyant des messages de grande taille.

Tableau 6-6 : Perte du débit utile en fonction du format de message

	Taille de l'entête	Somme des entêtes + redondance	Perte du débit utile
Message UDP de 47 Ko	64	1088	2,31 %
37 Messages UDP de 1300 octets	64	2368	5.05 %
37 Messages UDP de 1300 octets avec redondance (255,223)	64	2368 + 5888 = 8256	17.6 %

On peut conclure de cette expérience que notre application se doit de faire un compromis entre affichage (nombre d'images par seconde → taille des paquets de données) et robustesse (récupération de paquets perdus dans une image).

Expérience 3 : Perte de paquets dans différentes situations

Le troisième point intéressant à tester est le nombre de paquets perdus dans différentes conditions de proximité et d'interférence. Cette expérience que j'ai voulu proche

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

de celle d'une salle de classe, nous permet de connaître le pourcentage d'erreurs maximum et en moyenne.

Conditions d'expériences :

- ❑ Taille des données par message UDP : 1300 octets. Image de taille 48000 octets environ. Soit une quarantaine de messages par images.
- ❑ Délai inter-images : 500ms
- ❑ Délai inter-messages : 0ms
- ❑ Un message perdu → l'image est perdue

Nous faisons varier la distance entre l'émetteur et le récepteur. (Le PC émetteur étant un ordinateur de bureau, il a fallu traverser des murs afin d'accroître la distance). Nous récupérons le nombre d'images perdues pour 100 images.

Tableau 6-7 : Moyenne des pertes de paquets et nombre d'images perdues en fonction de la distance et des obstacles.

Distance (m)	1	2	4 + 1 mur	6 + 1 mur	10 + 2 murs	15 + 2 murs
Nombres d'images perdues	0	1	3	5	18	65
Moyenne perte de paquets par image	0	2	1	3	7	6

- ❑ A environ 10 mètres, l'état de la connexion entraîne un pourcentage d'images perdues de : $18 / 118 = 15.25\%$.
- ❑ A environ 15 mètres, l'état de la connexion entraîne un pourcentage d'images perdues de : $65 / 165 = 39.4\%$.

Nous supposons donc que pour une portée de 0 à 8 mètres, notre code correcteur d'erreurs permettrait de récupérer l'ensemble des images perdues. En effet, nous avons en moyenne une perte moyenne de 3-4 messages par images, soit environ 10% d'une image. Ce pourcentage est facilement récupérable grâce à un code correcteur d'erreurs.

Les résultats récoltés lors de mon expérimentation sont regroupés dans le tableau ci-dessous :

Tableau 6-8 : Même expérience avec l'envoi des images complètes dans les messages UDP

Distance (m)	1	6 + mur	8 + mur	12 + 2mur	15 + 2 mur
Nombres d'images perdues	0	4	10	49	67

Ces résultats sont semblables à ceux du tableau 5-5-7, sauf en ce qui concerne l'expérience à 12 mètres. Nous pouvons interpréter cette anomalie soit par des différences de conditions (orientation → modification de la propagation des ondes, expériences d'optique sur la même bande de fréquence ...), soit en supposant qu'on est à une distance où le protocole 802.11b diminue de moitié sa puissance d'émission.

Pour conclure sur cette expérience, on obtient le même taux d'erreur avec des images décomposées en messages UDP de taille plus ou moins petite. Cependant, avec des messages perdus de petite taille, il serait plus facile de récupérer l'image grâce à la redondance d'un code FEC.

Remarque sur les expérimentations

Pour cette phase d'expérimentation, je n'ai pu tester les performances face aux interférences. En effet, la société THALES possède des consignes de sécurité qui interdisent les équipements de type Wi-Fi ou Bluetooth dans l'enceinte des locaux (sauf dérogation). Une des conséquences est le délai à obtenir du matériel Wi-Fi pour mes expérimentations. Ce matériel m'aurait aussi permis de vérifier les performances d'un réseau 802.11g.. De même, j'aurai pu tester la conséquence d'interférences avec le protocole Bluetooth

Un autre problème survenu durant cette phase d'expérimentation est l'incompatibilité entre le logiciel d'audit du signal et l'ordinateur {Ord1}. Je n'ai donc pu enregistrer la puissance du signal.

6.2 Spécification

6.2.1 Réseaux

Le but de notre application réseaux étant essentiellement une diffusion d'images vers différents clients sur un réseau local, nous nous sommes tournés vers une application multipoint. Celle-ci est composée d'un ou plusieurs serveurs transmettant vers un ou plusieurs clients à travers des flux multicast UDP.

Un des points importants à spécifier dans une solution réseau est la gestion des erreurs et de flux. Pour cela, nous avons précédemment réfléchi à un code correcteur d'erreur qui éviterait toute retransmission à l'intérieur du réseau (i.e. toute surcharge). Cette solution qui laisse au client le soin de gérer le problème nous a aussi paru cohérente pour la gestion de flux. Notre idée est donc de laisser le client récupérer les images à la cadence qu'il souhaite sachant que l'émetteur quant à lui va envoyer à une cadence forte. Ainsi un PC lent pourra afficher seulement quelques images par seconde alors qu'un PC rapide lui, affichera avec une cadence maximum. La même réflexion s'applique pour un PC loin de l'émetteur et un PC très proche.

Pour le contrôle de congestion (toujours dans l'hypothèse d'une salle de classe ou de conférence), nous avons pensé qu'un retour d'information de contrôle des clients vers le serveur n'était pas nécessaire. Notre application se basant sur le protocole de transfert UDP, elle ne réduira pas son débit en fonction de la congestion du réseau. Notre choix ne semble pas incohérent sur un réseau local dédié à notre application.

Finalement, on aura :

- ❑ un émetteur envoyant des images avec un certain taux de redondance et à une cadence élevée
- ❑ des récepteurs récupérant les images à une cadence plus ou moins faible (spécifiée par la distance à l'émetteur, les interférences, la charge du réseau et bien sûr le temps de traitement des images) et essayant de reconstruire le plus d'images possible (grâce au code correcteur d'erreur).

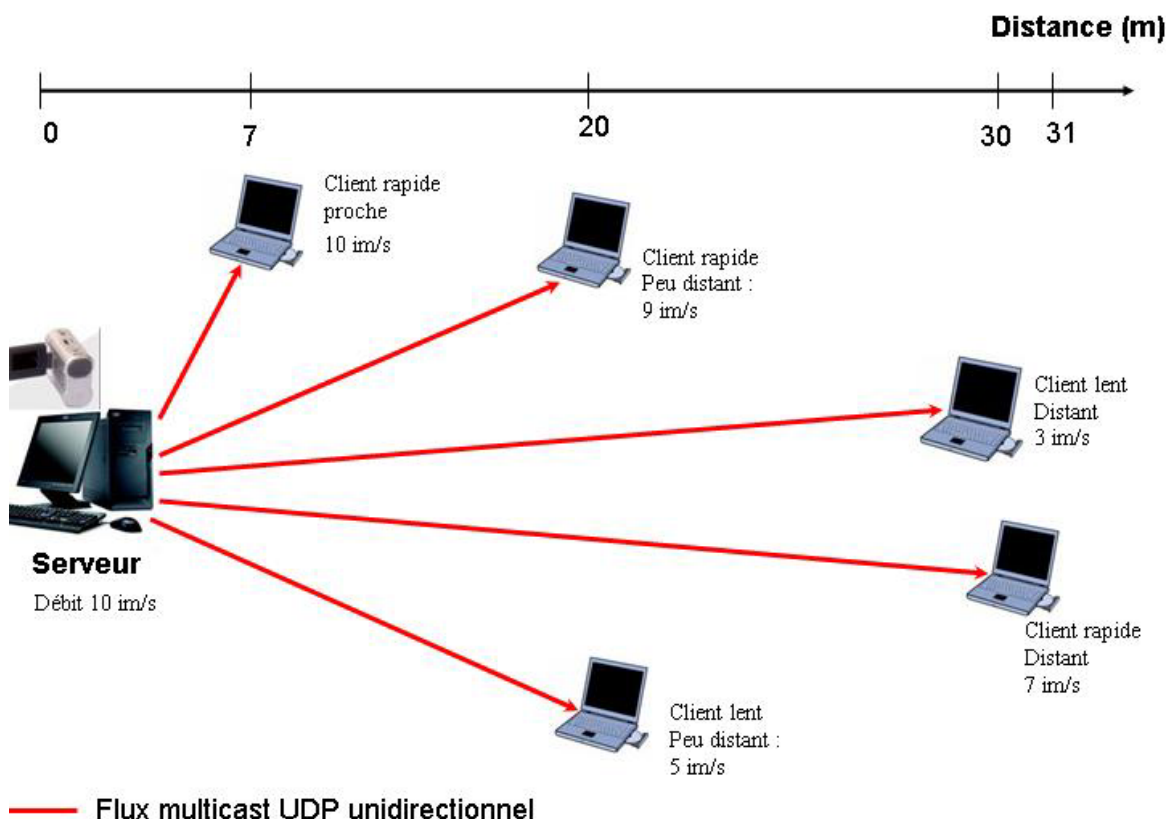


Figure 6-6 : Schéma explicatif du principe d'émission. Le serveur émet et les clients essaient de récupérer le plus d'images possible selon leur rythme (problème de distance ou de temps de traitement). Le nombre d'image par seconde n'étant pas un paramètre critique.

En fonction des résultats précédents sur les composants multicast que nous utilisons, nous avons décidé de créer deux modes :

Mode rapide

Le premier permet d'envoyer les images une par une dans des messages multicast UDP sans code correcteur d'erreur. Cette solution accroît les performances dans des réseaux de bonne qualité (sans interférence, avec les récepteurs proches de la station émettrice). Ainsi, on se rapproche d'une situation avec une salle de classe assez petite, un petit nombre d'élèves utilisant PortaNum™ et un réseau Wi-Fi dédié à la diffusion des images PortaNum™. Cette solution nécessite de fournir des images de tailles comprises entre 1 500 et 65 000 octets (limite définie par le format UDP). Il suffit pour cela de faire varier le taux de compression pour obtenir des tailles d'images comprises entre ces grandeurs.

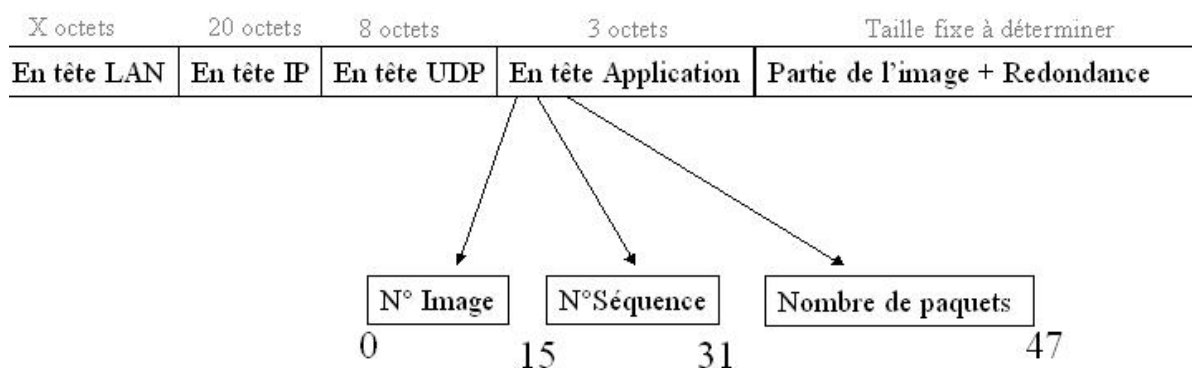
PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Mode sécurisé

L'autre solution permet de recouvrir un grand nombre d'erreurs de paquets. En effet, avec des paquets de petites tailles et un entête numérotant les images, les paquets et le nombre de paquets, on peut déterminer quel paquet a été perdu et ainsi le récupérer plus facilement. Des paquets de tailles réduites permettent dans le cas d'une simple perte de facilement recomposer le paquet perdu grâce au code Reed Solomon.

L'objectif de ce mode est de garder un fonctionnement correct dans des amphithéâtres (grande distance entre émetteur et récepteur), lorsque beaucoup d'élèves utilisent PortaNum™ (salle de classe dans des établissements spécialisés) ou lorsque le matériel informatique (routeur, PC, modem) sert à différentes applications.

Ce dernier mode donne la spécification de format de message suivant :



*X dépend du protocole de transport

Figure 6-7 : Format final des messages transmis sur le réseau par notre application

6.2.2 Traitement et compression des images

Les images capturées par les caméras numériques sont au format Bitmap. Sans compression, une image de résolution 720 x 576 prend environ 1.2Mo, il semble donc peu intéressant d'envoyer l'image sans compression. Le choix de compression de notre flux d'image s'est finalement posé entre le M-JPEG et le MPEG :

- Le principe du **Motion JPEG** (noté MJPEG ou M-JPEG) consiste à appliquer successivement l'algorithme de compression JPEG aux différentes images d'une séquence vidéo. Son débit comme vu précédemment est de 8 à 10 Mbit/s.
- Le principe du MPEG (*Moving Pictures Experts Group*) est d'exploiter l'idée de **redondance temporelle** (d'une image à une autre, il n'y a que peu de modifications). Ainsi, ce codage va transmettre une première image complète puis le changement d'une image à l'autre. Il existe différents standards MPEG-1, MPEG-2, MPEG-4, MPEG-7, MPEG-21 pouvant diminuer le débit au-delà de 1Mbit/s pour une résolution identique. Le MPEG-4 est une technologie qui est soumise à une licence, ce qui rend son implémentation plus longue (délai d'obtention de la licence)

Tableau 6-9 : Caractéristiques des différents formats de compression

	MPEG-1	MPEG-2	MPEG-4	T-JPEG
Résolution vidéo maximale	352 x 288	1920 x 1152	720 x 576	Aucune
Débit standard	1380 kbit/s (352 x 288)	6500 kbit/s (720 x 576)	880 kbit/s (320 x 240)	10 Mbit/s (640 x 480)
Nombre d'images par seconde	25	25	25	25

Finalement, nous avons choisi la solution du M-JPEG. La première raison vient du fait que les traitements (contraste, contour, ...) font apparaître les défauts d'une image trop comprimée. De plus, on considère que, dans notre application, la qualité de l'image transmise est plus importante que le nombre d'images comme l'ont souligné les utilisateurs. En effet, durant nos différentes expérimentations, les enfants ont mis en évidence le fait qu'un rafraîchissement trop important n'était pas utile pour la prise de note et que de travailler sur une cadence de 3 images par seconde leur convenait parfaitement. Avec cette

information, il nous a paru plus intéressant pour l'instant d'envoyer des images de grande résolution et facilement adaptable à différents formats :

- ❑ 640 x 480 → webcam
- ❑ 720 x 576 → caméra numérique
- ❑ N'importe quel format d'image → Disque dur

Nous pourrions même envoyer des images de résolution variable adaptées au tableau mis en place dans l'école, en nous basant sur des caméra IP ou des caméras dédiés à la vidéo surveillance. Une autre idée serait d'envoyer à chaque émetteur seulement la région d'intérêt¹³ et non plus toute l'image. Un autre point en faveur de la compression M-JPEG est sa simplicité d'utilisation, sa grande compatibilité ainsi que la grande qualité d'image¹⁴.

Lors de la transmission d'un flux vidéo, afin de garder une taille d'image peu variable, nous avons pensé adapter le taux de compression d'une image à une autre pour que les tailles restent quasi identiques.

Une modification importante sur la version de base est de faire par défaut un filtrage permettant ainsi de gagner en taille sur le résultat d'une compression JPEG. En effet, les images déjà filtrées possèdent moins de bruit et sont après compression d'une taille plus faible.

De même, la solution de compression JPEG2000 a été envisagée car elle permettrait de gagner un facteur 2 sur la taille des images après compression et cela sans perte de qualité, ce qui n'est pas négligeable du point de vue de notre application. Le problème rencontré est celui de la licence JPEG2000 qu'il faut obtenir ainsi que de l'évaluation du temps de compression par rapport à du JPEG normal. Cette évaluation sur une implémentation a été faite et nous avons trouvé un temps de compression d'environ 150 ms pour une résolution de 640 x 480. Ces résultats nous ont incité à repousser l'intégration de cette solution dans la version en cours de développement de PortaNum.

6.2.3 IHM

L'application PortaNum comme toute solution réseau nécessite de réfléchir aux deux interfaces, celle du client et celle du serveur. Puisque nous voulions que le serveur puisse

¹³ Région d'intérêt est la partie de l'image que l'utilisateur voit en se moment. Un traitement sur une région d'intérêt permet de ne pas traiter toute l'image.

¹⁴ http://www.axis.com/products/video/about_networkvideo/mpeg4.htm

envoyer des images prétraitées, les interfaces client et serveur sont identiques à la version monoposte. Seuls les paramètres d'émission et de réception différencieront ces interfaces. Finalement, nous avons le choix de développer :

- ❑ Deux applications quasi semblables.
- ❑ La même application où le choix entre client et serveur se situe dans le menu expert.

Les deux cas requièrent la réflexion sur les besoins spécifiques de l'interface du côté serveur et du côté client.

6.2.3.1 IHM serveur

Nous pensons offrir à l'utilisateur les différents paramètres suivants :

- ❑ Le choix de l'adresse IP du groupe multicast (sachant qu'une valeur par défaut sera partagée par le client et le serveur).
- ❑ Le choix entre les deux modes : envoi d'image par image ou une image découpée en plusieurs messages.
- ❑ La cadence d'envoi de deux images.
- ❑ La compression JPEG ou JPEG2000 ainsi que le taux de compression.
- ❑ Le pourcentage de redondance dû au code correcteur d'erreur (ce paramètre n'est pas encore mis en place car il nous faut de plus amples expériences pour être sûr de son utilité).

6.2.3.2 IHM client

Le client pour bien gérer la réception a aussi besoin de connaître le mode de transmission, l'adresse IP, le port, etc ... Nous avons plusieurs choix pour que le serveur communique ses paramètres aux clients :

- ❑ Garder l'interface de la version monoposte et trouver un moyen de communiquer ces différentes spécifications au lancement du réseau. Par exemple, on pourrait imaginer que le serveur envoie toutes les secondes un message broadcast contenant tous les paramètres d'émission. Une autre solution serait de créer une liaison TCP bidirectionnelle pour envoyer des messages de contrôle (il faudrait quand même communiquer l'adresse IP au

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

départ). La dernière possibilité serait de faire passer ces messages de contrôle dans la bande, c'est à dire dans le flux multicast en le séparant par un flag dans l'entête.

- Remettre tous les choix du serveur au niveau du client, laissant ainsi la responsabilité aux utilisateurs de se communiquer les paramètres quand ils sont différents de ceux par défaut. Dans la plupart des cas, cette gestion pourra être mise en place dans chaque classe ou amphithéâtre par la personne responsable du réseau puis les paramètres d'installation conservés dans le fichier d'initialisation de PortaNum.

Nous avons décidé après réflexion et pour simplifier la mise à jour des versions de regrouper la version monoposte et la version client (avec seulement un paramètre de lancement différent). Cette version sera celle disponible sur le site Internet et mise en place dans les écoles dès la rentrée prochaine.

Les quelques différences dans l'interface par rapport à la version monoposte se trouvent dans un onglet de la boîte de dialogue expert (permettant le choix des paramètres réseaux) mais aussi dans le bouton permettant de passer du mode de capture vers le mode réseau et situé dans la barre de menu. De même, la possibilité de mettre en pause la réception réseau sera laissée à l'utilisateur lorsque celui-ci voudra prendre des notes sur une image fixe.

Finalement, une caméra réseau s'utilisera comme n'importe quelle caméra numérique rendant la gestion du réseau transparente à l'utilisateur final.

6.3 Architecture

6.3.1 Emission

6.3.1.1 Cadencement

L'architecture d'émission part du principe suivant : l'utilisateur définit lui-même la cadence à laquelle il veut transmettre les images. Il va donc capturer et envoyer les images toutes les X millisecondes, X étant un paramètre d'émission. Cette solution permet au serveur de voir le même rafraîchissement qu'un client ne subissant aucune erreur de paquet ou d'image.

6.3.1.2 Modes d'émission

Comme nous l'avons vu au chapitre précédent, nous avons décidé d'établir deux modes de transmission:

- **Rapide** : qui envoie une image complète dans des messages UDP.

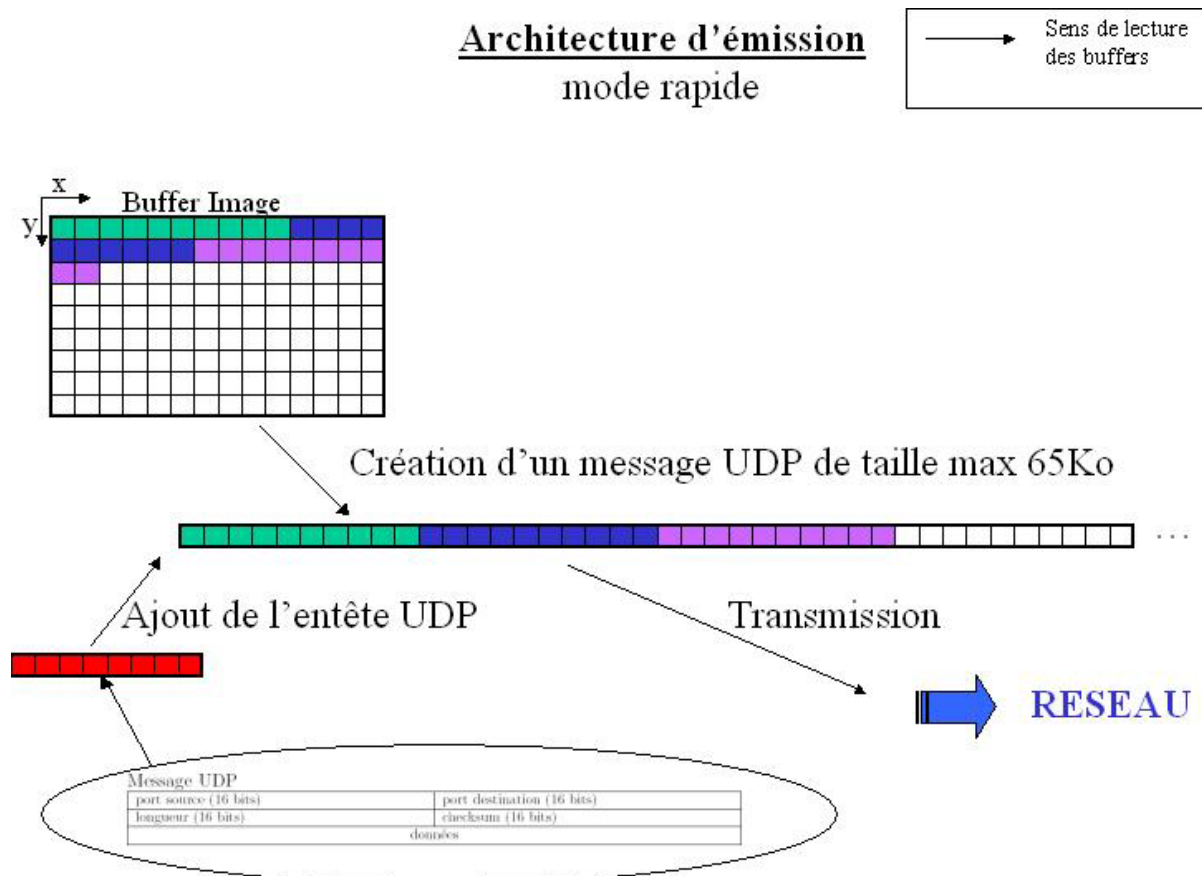


Figure 6-8 : Dans le mode rapide, l'émetteur va simplement recopier le buffer dans un message UDP en ayant vérifié au préalable qu'il respecte la taille maximale de 65Ko. (En effet, la longueur est codée sur 16 bits soit au maximum 2^{16} bits.

- **Sécurisé** : qui envoie les images dans des paquets UDP de petite taille (1300 octets) et dans le désordre afin de mieux recouvrir, grâce au code correcteur d'erreur, la perte d'un ou plusieurs paquets.(cf 6.3.3)

Le choix du mode est laissé à l'utilisateur dans l'interface graphique.

6.3.2 Réception

Réception d'un message UDP

L'architecture de réception doit partir de l'idée suivante : le numéro de séquence détermine l'ordre d'émission des paquets mais aussi la place dans l'image. Ainsi, sans se préoccuper d'entrelacement, l'image est considérée comme un ensemble de blocs de taille fixe. De cette manière, le récepteur va utiliser le numéro de séquence pour placer les données au bon endroit dans le buffer de réception. Parallèlement, il va mettre un flag pour signaler que cette partie de l'image a été correctement reçue.

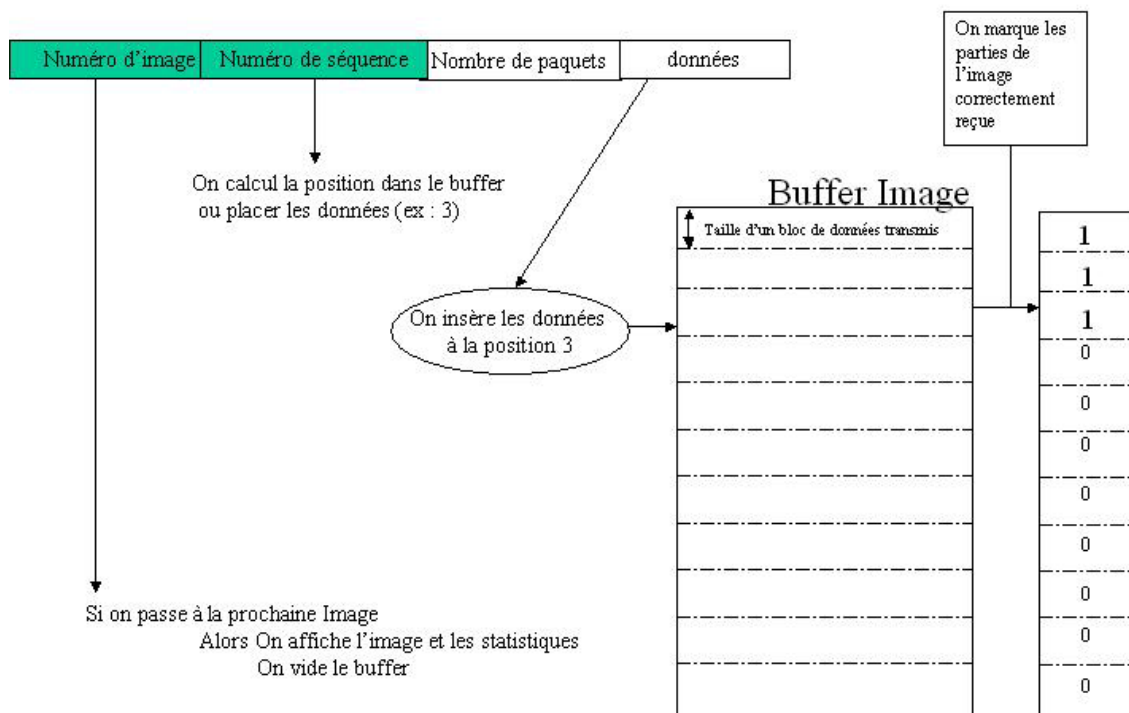


Figure 6-9 : Fonctionnement de la réception d'un message

Finalement, si on a récupéré correctement tous les paquets de données, on va afficher l'image et vider le buffer. Dans le cas contraire, on va grâce au flag remplir les parties de

l'image perdue et essayer grâce à un code Reed-Solomon de reconstruire l'image. Il est à noter que grâce aux flags de réception, on détermine la place des pertes de paquets augmentant la capacité de recouvrement d'erreur d'un facteur 2.

Gestion de la fluidité

Comme dans tout affichage de flux vidéo, il est essentiel d'avoir une certaine fluidité dans la fréquence d'affichage des images. Un récepteur, ne pouvant pas se synchroniser sur la fréquence d'émission des images pour des raisons de distance au serveur, de gigue, ou de temps de traitement des images, va rater un certain nombre d'images. Ainsi, pour conserver un bon rafraîchissement des images, il va appliquer un algorithme permettant d'évaluer le délai d'affichage entre deux images en se basant sur son expérience passée. L'utilisateur aura le choix de se servir ou non de cet algorithme sachant que par défaut il ne sera pas exécuté.

Cet algorithme fonctionne de la manière suivante :

- Le récepteur conserve le nombre d'images bufférisées X depuis le dernier affichage et en fait une moyenne avec la formule suivante :
 - **Moyenne $T = \alpha \times X + (1-\alpha)\text{Moy}(T-1)$**

Avec α : variable compris entre 0 et 1 déterminant l'importance de la dernière expérience par rapport aux autres.

- Le récepteur utilise la moyenne qu'il compare avec deux seuils : un minimum et un maximum. Si la moyenne dépasse le seuil maximum, il va diminuer le délai d'affichage tandis que si elle passe en dessous du minimum, le délai va augmenter, espérant ainsi tendre vers la meilleure fréquence d'affichage.

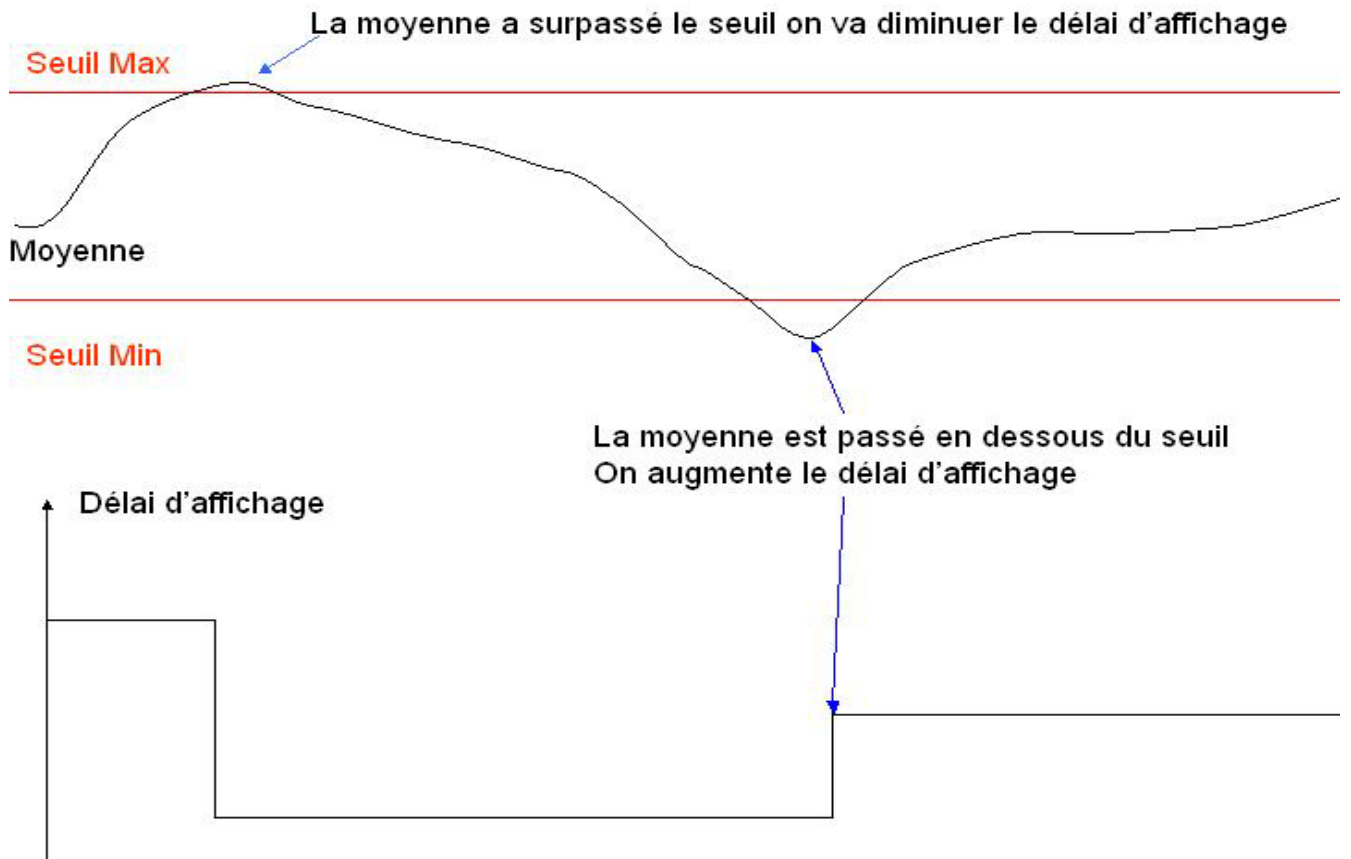


Figure 6-10 : On voit ici la correspondance entre la moyenne calculée et le délai d'affichage entre deux images. En effet, si la moyenne dépasse un certain seuil maximum ou minimum, on va modifier la valeur du délai.

6.3.3 Compression et décompression

Dans cette partie, nous allons voir schématiquement comment fonctionnent du côté serveur la compression et l'entrelacement puis du côté client, la décompression et le désentrelacement. On va aussi différencier les cas de réception avec et sans erreurs.

Côté Emetteur

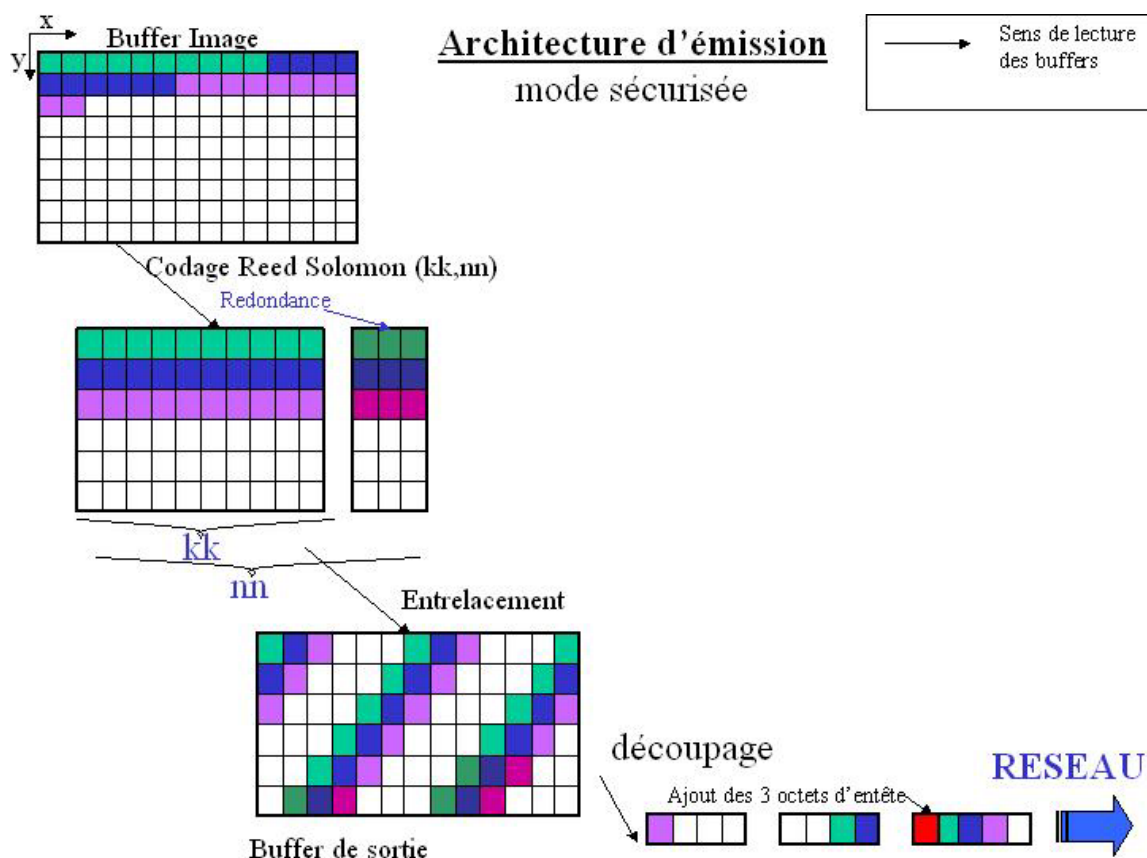


Figure 6-11 : L'architecture d'émission dans le mode sécurisé

Comme on peut le voir sur ce schéma, l'émission commence par utiliser le codage Reed Solomon sur le buffer contenant l'image à transmettre. Cet encodage se décompose en 2 étapes :

- On découpe l'image en mots de code, i.e des morceaux de données de taille fixe nommé kk . On récupère le nombre de mots de code ainsi fabriqués, appelé $nBlocks$. Le mot de code courant est appelé $iBlock$.
- On rajoute pour chaque mot de code, une redondance obtenue par l'algorithme de Reed Solomon. On obtient alors nn bits pour kk bits de données.

Ensuite, on va entrelacer ces mots de code à l'intérieur du buffer de sortie. La position du bit courant i dans le nouveau buffer sera déduite de la formule suivante :

$$\square \text{ nouvellePosition} = iBlock + nBlock \times i$$

Côté récepteur

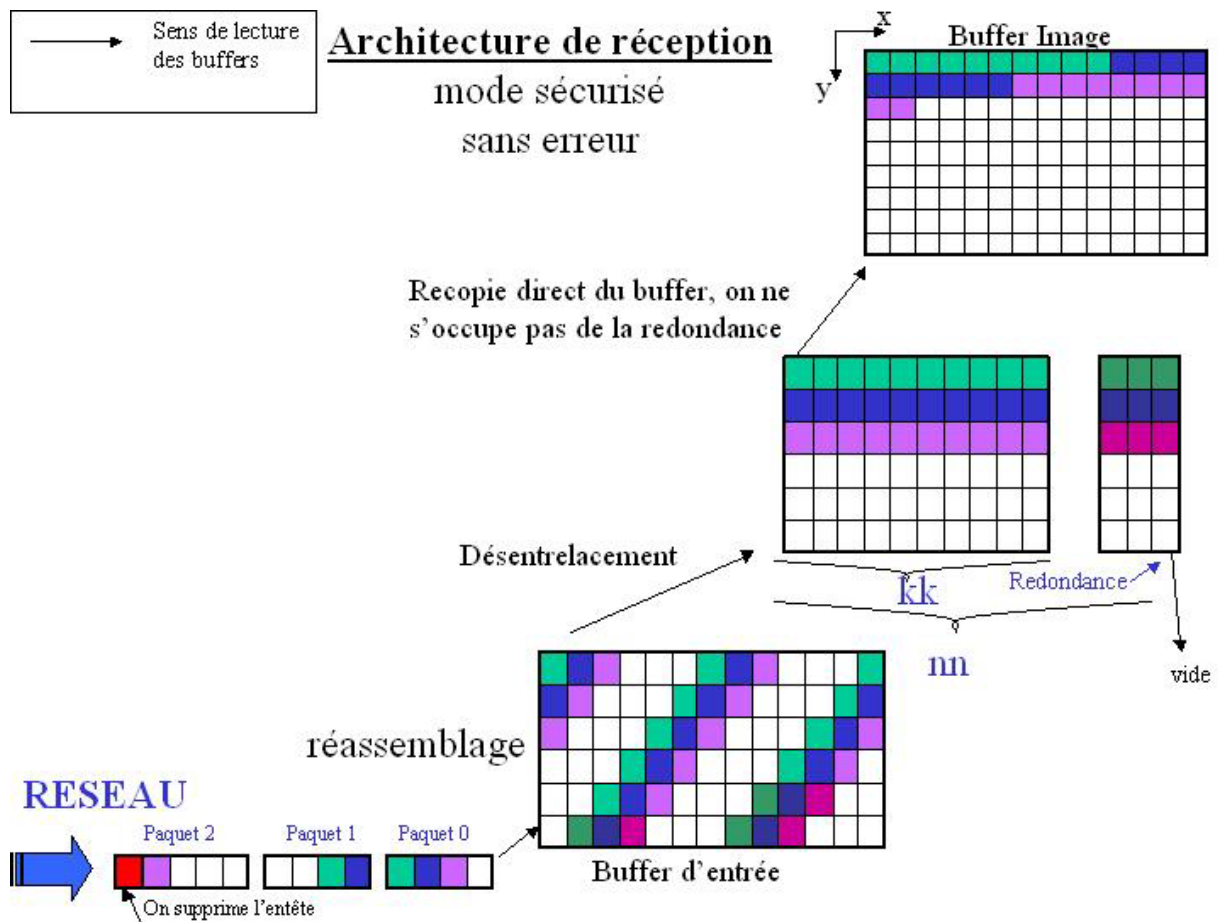


Figure 6-12 : Architecture de réception lorsque le récepteur n'a trouvé aucune erreur de paquet.

Comme nous l'avons vu précédemment lors de la réception, si un paquet est reçu correctement, on va mettre un flag. Si tous les flags sont bien positionnés, on considère avoir

reçu l'image sans erreur. Dans ce cas précis, on va entrelacer puis directement recopier vers le buffer de sortie. En effet, il serait inutile de décoder le Reed Solomon puisqu'il suffit de récupérer l'image et de supprimer la redondance.

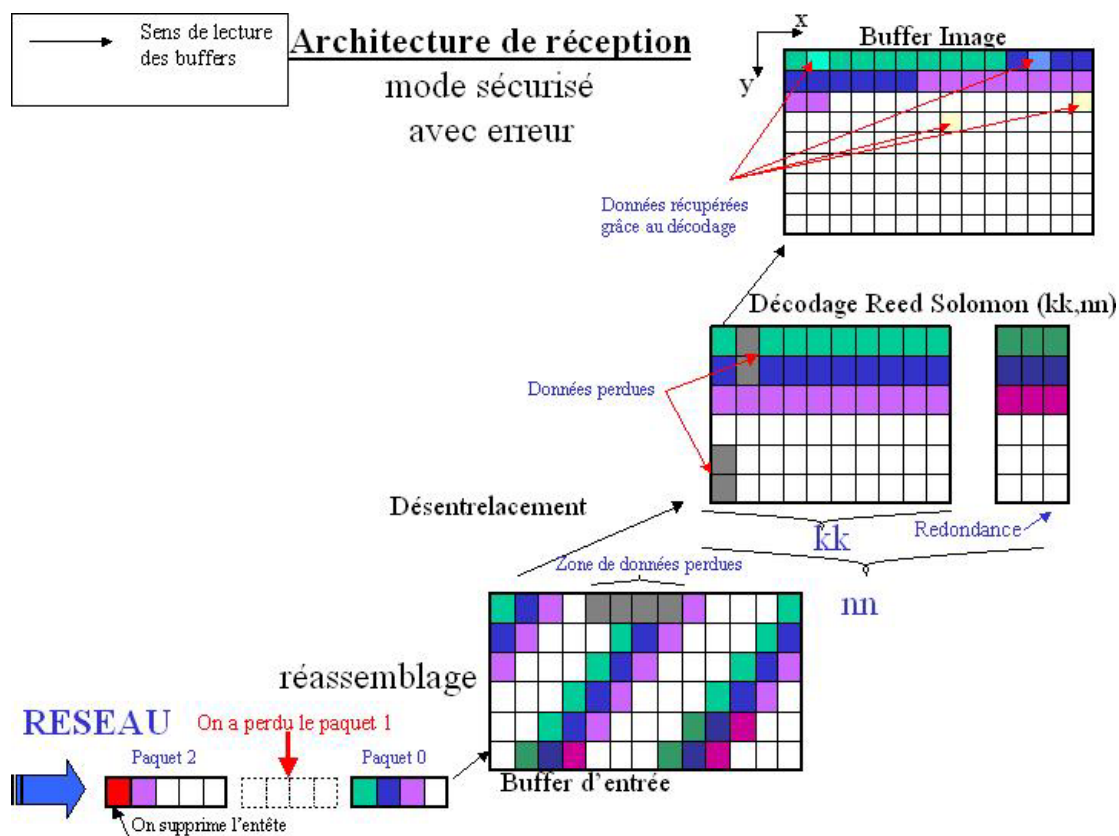


Figure 6-13 : Architecture de réception dans le cas où une perte de paquet est remarquée.

Cependant, il peut arriver que le passage par le réseau entraîne une perte de paquet. Dans ce cas, notre récepteur va désentrelacer les différents paquets reçus puis récupérer les données perdues et dispersées dans le buffer grâce à la redondance. La taille de la redondance ($nn - kk$) détermine le nombre de données que l'on va pouvoir récupérer grâce à l'algorithme de Reed Solomon.

Pour utiliser cette information de la manière la plus pertinente possible, notre récepteur utilise une condition simple de traitement lui permettant de ne pas perdre de temps à décoder s'il est sûr de ne pas pouvoir récupérer l'image :

- Si le nombre de paquets perdus > 5 (valeur que nous avons déterminée après plusieurs tests)
 - Alors on ne décode pas l'image, on la considère comme perdue.

Pour conclure, la description de notre architecture client/serveur permet de comprendre plus facilement nos deux orientations :

- La performance : On ne fait aucun algorithme de recouvrement d'erreur, on compresse simplement l'image puis on l'envoie afin de ne pas perdre de temps dans la recopie des buffers ni dans l'implémentation des algorithmes.
- La sécurité : On ajoute de la redondance en utilisant le code Reed Solomon auquel on ajoute l'entrelacement des images. Ainsi, on peut récupérer jusqu'à 5 pertes de paquets par image (d'environ 50ko) même si pour cela on perd du temps à modifier les différents buffer.

6.4 Version actuelle

6.4.1 Interface graphique

Après plusieurs expérimentations et dans le but de bien évaluer nos différents paramètres dans les futures installations, nous avons établi l'interface graphique suivante pour le serveur :

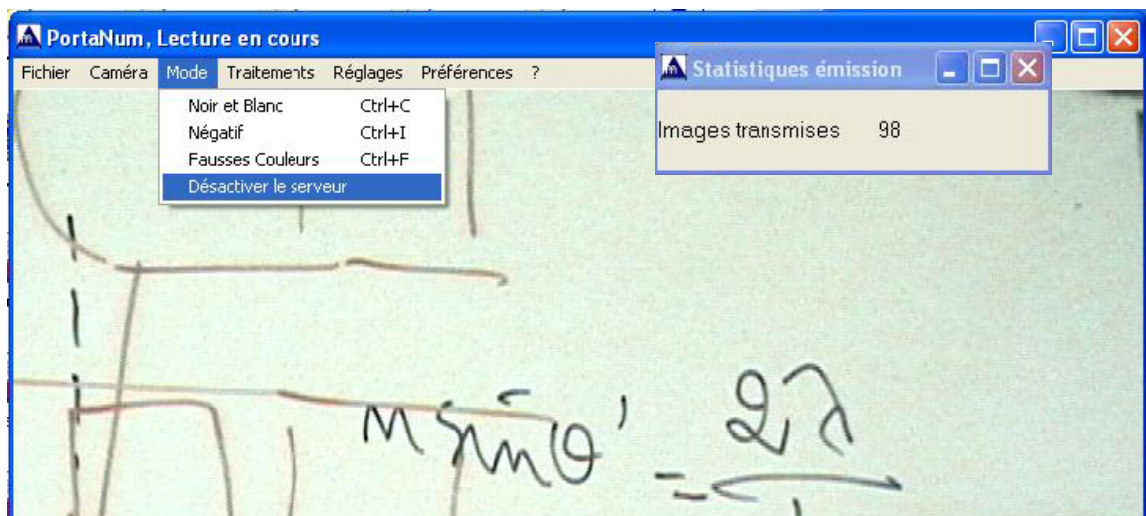


Figure 6-14 : L'interface du serveur. On peut voir le bouton permettant au serveur d'arrêter la transmission des images sur le réseau tout en laissant la capture active. Les statistiques réseaux permettent de vérifier le bon fonctionnement de l'émission et de visualiser la cadence d'émission.

Le bouton dans le menu mode permet au serveur d'arrêter la transmission. En effet, un professeur peut vouloir parcourir des images sans les diffuser à ses élèves.

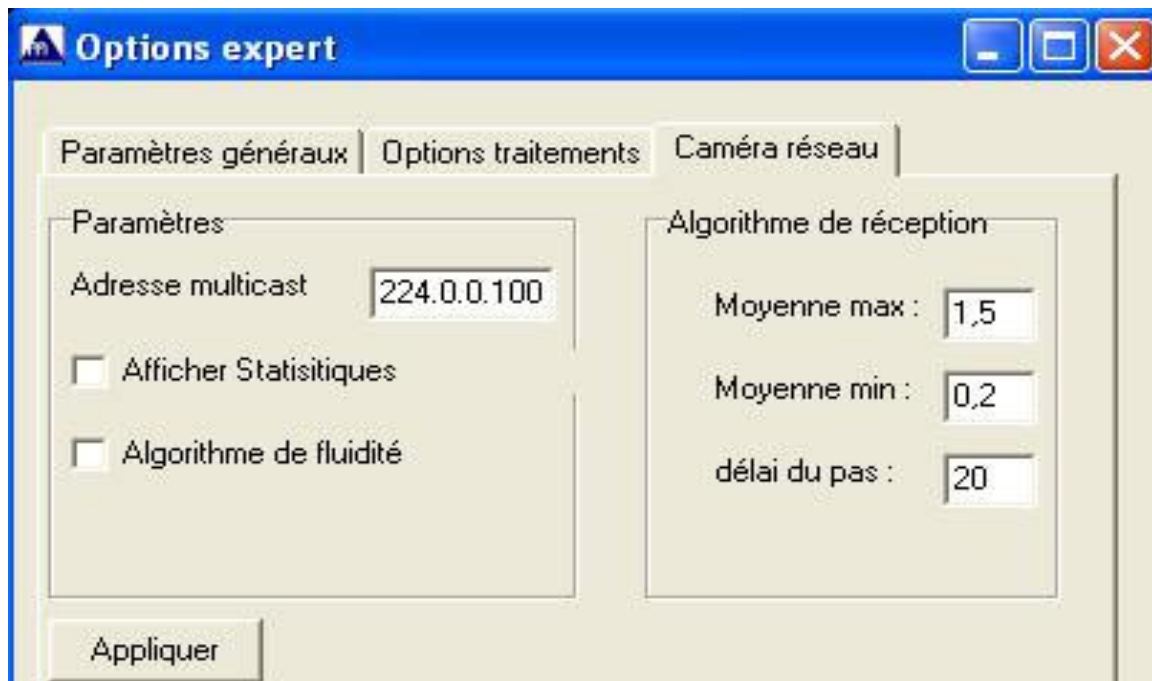


Figure 6-15 : Capture d'écran du menu expert dans la version client. On remarque les paramètres de base ainsi que les paramètres de l'algorithme d'affichage. Une case à cocher permet de visualiser les statistiques réseaux.

Le client lui n'a pour l'instant que le choix de l'adresse IP. Ainsi, si on lance plusieurs applications client sur une même machine, on peut visualiser plusieurs flux vidéo, chacun déterminé par un port et une adresse IP différentes. Les paramètres de l'algorithme de cadencement sont laissés dans cette version. En effet, le peu de situation testée ne nous a pas encore permis de trouver des optimums pour les paramètres dans différentes situations.



Figure 6-16 : Fenêtre d'affichage des statistiques renseignant le nombre d'images correctement reçues. Ainsi que le débit et la fréquence d'images par seconde.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

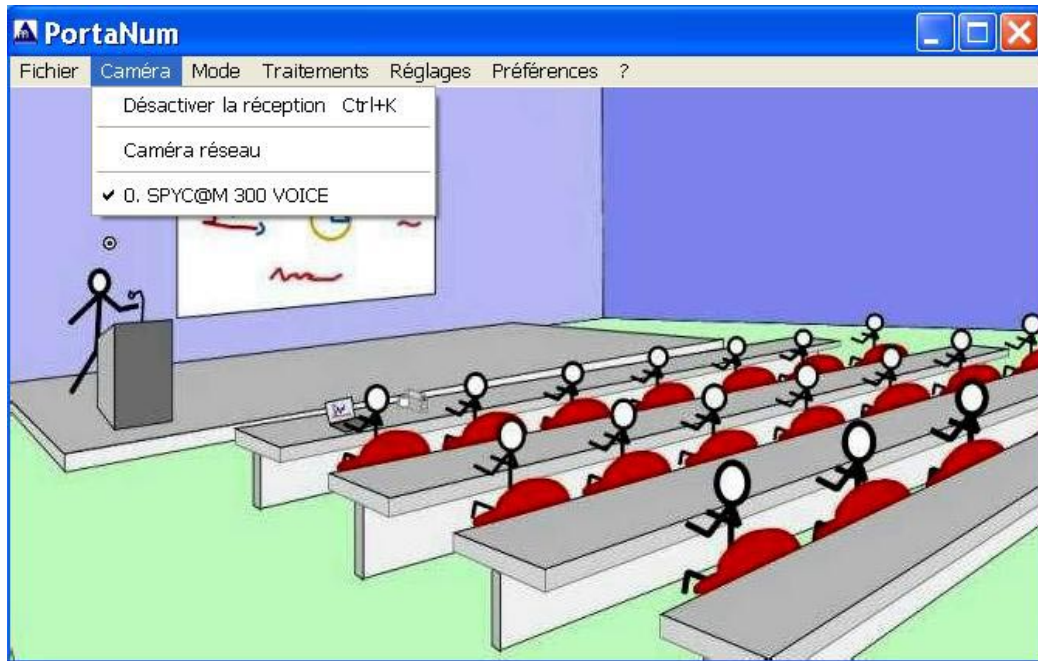


Figure 6-17 : Le client peut utiliser de manière intuitive et comme toute autre caméra, le flux vidéo.

Le bouton caméra réseau situé dans le sous menu « Caméra » permet d'utiliser le flux vidéo réseau comme n'importe quelle autre caméra. C'est-à-dire, permettant de passer d'une caméra à une autre facilement ou de mettre en pause ou redémarrer la caméra réseau.

6.4.2 Résultats d'expérimentations

Performance de la version monoposte

Dans cette partie, nous avons mesuré plusieurs délais en fonction du traitement appliqué sur différents ordinateurs.

Tableau 6-10 : Délai d'affichage et de traitement en fonction du type de traitement sur des images 640 x 480 sur {Ord1}

Types de traitement	Temps d'affichage (ms)	Temps de traitement (ms)
Automatique	2.3	10.8
Scène sombre	2.4	10.6
Contrastes de lumières	2.9	19.5
Texte imprimé	2.4	11.3
Texte manuscrit	2.4	14.5

On remarque que dans la plupart des cas le délai de traitement plus le délai d'affichage n'excède pas les 20 ms. Notre application peut alors afficher 50 images par seconde, on omet dans ce calcul le temps d'acquisition.

Tableau 6-11 : Délai d'affichage et de traitement en fonction du type de traitement sur des images 720 x 576 sur {Ord1}

Types de traitement	Temps d'affichage (ms)	Temps de traitement (ms)
Automatique	2.9	12.4
Scène sombre	3.1	12.6
Contrastes de lumières	3.3	24.4
Texte imprimé	2.9	12.5
Texte manuscrit	2.9	12.8

On remarque qu'avec une résolution supérieure on perd 8 ms au niveau du temps de traitement et d'affichage. Ce qui nous donne une fréquence maximum de 33 images par seconde.

Tableau 6-12 : Délai d'affichage et de traitement en fonction du type de traitement sur des images 640 x 480 sur {Ord2}

Types de traitement	Temps d'affichage (ms)	Temps de traitement (ms)
Automatique	30	60
Scène sombre	35	60
Contrastes de lumières	40	105
Texte imprimé	35	70
Texte manuscrit	32	68

On remarque que sur un ordinateur beaucoup plus lent les temps d'affichage et de traitement ne permettent qu'un rafraîchissement d'environ 8 images par seconde.

Expérimentation sur la version réseau

Expérience de l'ordinateur {Ord1} vers {Ord2} en passant par un routeur. Celui-ci reliant {Ord2} avec une connexion sans fil 802.11g et {Ord1} avec la connexion filaire (intranet à 1 Go/s). {Ord1} est le serveur, il envoie les images de la caméra numérique au format 720 x 576.

Nous avons aussi mesuré quelques délais sur la version réseau (grâce à la version mise à jour de UDPTools) que nous détaillons dans le tableau ci-dessous :

Tableau 6-13 : Différents délais en fonction du mode d'émission

Mode	Codage	Décodage	Envoi	Réception
Sécurisé	8	15	8	38
Rapide	0	0	3	7

On remarque qu'en mode sécurisé le temps de traitement de l'émission, réception, codage et décodage est de 69 ms alors qu'en mode rapide, il n'est que de 10 ms. La conséquence en nombre d'images par seconde est :

- ❑ En mode sécurisé : On a $35 + 20 = 89$ ms (temps de réception + affichage, en considérant que la réception et l'émission se font en parallèle) → 11.3 images par seconde.
- ❑ En mode rapide : On a $10 + 20 = 30$ ms → 33 images par seconde possible.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

En mode sécurisé, sur ma machine connectée à une caméra numérique capturant des images 720 x 576, mon serveur arrive à transmettre au maximum 9 images par seconde. On peut l'expliquer par le fait que le serveur a besoin de capturer l'image, d'appliquer un traitement, de la compresser, de la découper en lui rajoutant des entêtes puis de la transmettre via le réseau sans fil. Ce qui fait un temps de calcul trop important pour envoyer à la cadence maximum du réseau.

Tableau 6-14 : Fréquence d'affichage et taux de perte en fonction de la distance avec une émission de 2 images par seconde.

Distance (m)	1	6	10	15	25	30	35	40
Fréquence (Im/s)	2.0	2.0	2.0	2.0	1.98	2.0	1.98	1.97
Perte d'images sur 100	0	0	1	1	3	1	7	4
Taux de perte	0	0	1%	1%	3%	1%	7%	4%

PortaNum dans son mode sécurisé atteint des taux d'erreurs négligeables pour l'utilisateur. De plus, j'ai pu remarquer que de nombreuses pertes de 1, 2, 3 ou 4 paquets à la suite ont été récupérés grâce au code correcteur d'erreur. Je n'ai pu quantifier une moyenne des pertes de paquets par faute de temps.

Par la suite, j'ai voulu faire un graphique permettant de quantifier le taux d'erreur en fonction de l'utilisation ou non d'un code correcteur d'erreur. Cette expérimentation se fait entre les ordinateurs {ord5} et {Ord6} connectés grâce au {R3}. On fait varier la distance en intérieur sans obstacle direct (seulement des murs de type préfabriqués).

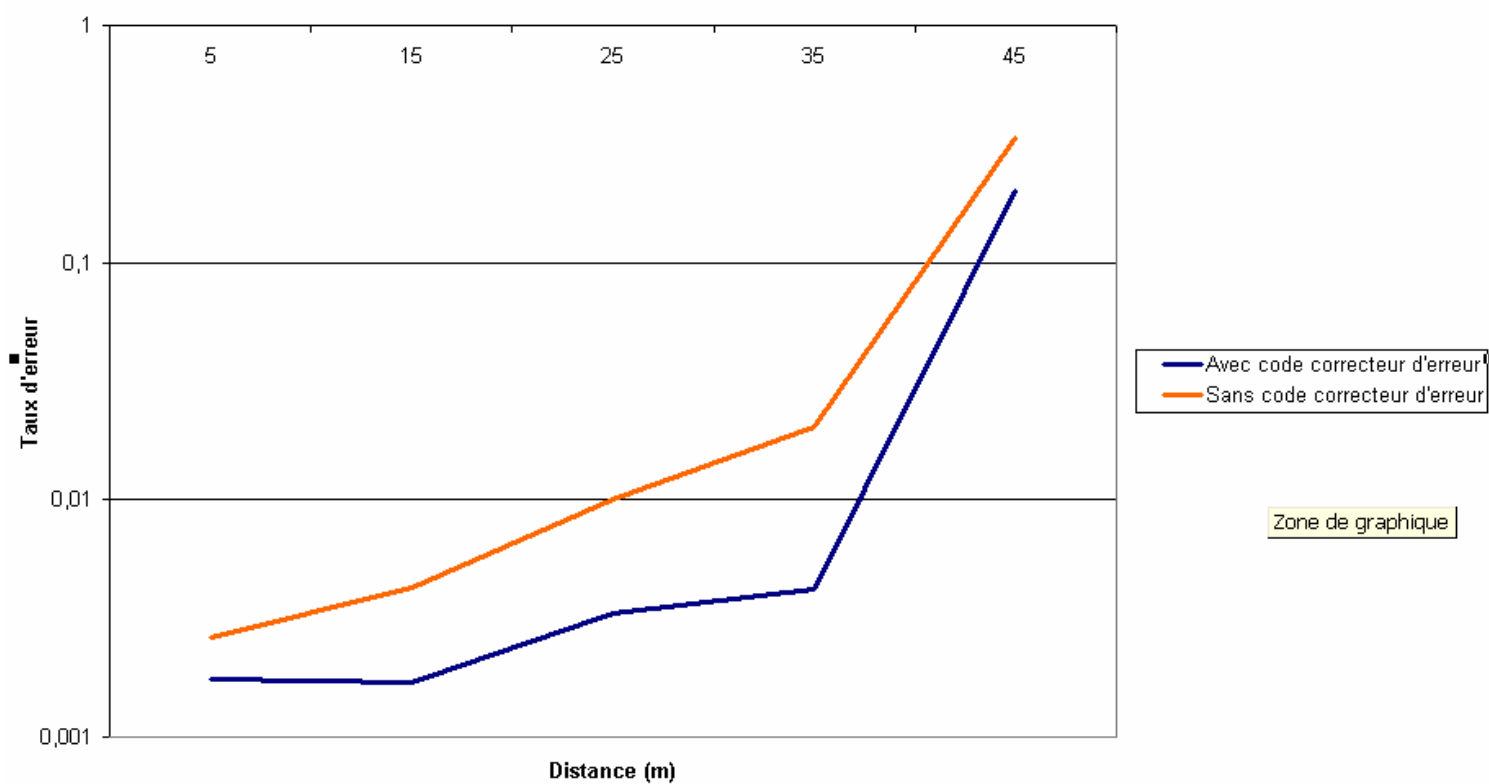


Figure 6-18 : Taux d'erreur en fonction de la distance et de l'utilisation d'un code correcteur d'erreur

Ce graphique nous montre bien une Diminution du nombre de pertes grâce au code correcteur d'erreur. De plus, si notre application supportait environ une image perdue sur 100 alors on pourrait gagner environ 15 mètres entre le client et le serveur grâce au code correcteur d'erreur.

Autonomie de la nouvelle version

Finalement, notre application visant essentiellement à être installée sur des ordinateurs portables, j'ai trouvé intéressant de vérifier la puissance consommée par notre logiciel dans différents cas d'utilisation. Ces résultats obtenus sur des ordinateurs neufs {Ord5} dont la batterie est en parfait état, et faits au début du déchargement pour chaque expérience devraient nous donner des estimations donnant une bonne idée de l'autonomie d'un ordinateur utilisant PortaNum.

Tableau 6-15 : Autonomie des ordinateurs portables dans différents cas d'utilisation

Type d'utilisation	Aucune acquisition	Réception d'un réseau Wi-Fi		Capture locale Webcam	
		2	5	15	25
Fréquence image (im/s)	/	2	5	15	25
Puissance consommée (W)	19	27	35	35	39
Autonomie estimée (h)	2h45	2h	1h30	1h30	1h20

Les résultats reportés dans le tableau ci-dessus montrent qu'il y a un compromis à faire entre autonomie et fréquence de rafraîchissement des images que l'on soit en mode capture locale ou en réception sur le réseau. En effet, en mode réseau si on décide de ne traiter que 2 images au lieu de 5 (comme nous le permet notre algorithme de fluidité), on peut gagner jusqu'à 30 minutes (Soit 18% de l'autonomie globale).

6.5 Déploiement d'un réseau WI-FI

6.5.1 Portée

Dans le cas d'une installation Wi-Fi, nous allons rappeler ici quelques détails importants à connaître à propos des portées du réseau et des interférences possibles.

Tableau 6.16 : Portée d'un réseau 802.11b en milieu intérieur.

Débit (en Mbit/s)	Portée (en mètre)
11	50
5.5	75
2	100
1	150

Tableau 6.17 : Portée d'un réseau 802.11g en milieu intérieur.

Débit (en Mbit/s)	Portée (en mètre)
54	10
48	17
36	25
24	30
18	40
12	50
9	60
6	70

On peut remarquer que la portée des réseaux est une fonction décroissante du débit du réseau. Ces valeurs qui sont dépendantes de beaucoup de facteurs et qui ne sont pas

forcément tout le temps valable, nous donnent quand même un indice des attentes que l'on peut avoir de notre réseau.

Il est à noter qu'en extérieur, l'utilisation d'un réseau de type 802.11a est interdite car l'utilisation de la bande de fréquence de 5GHz est interdite.

En extérieur, pour un réseau 802.11b, la portée augmente considérablement. La cause en est bien sûre l'absence d'obstacles gênant la transmission des ondes radios.

Tableau 6.18 : Portée en extérieur d'un réseau 802.11b.

Débit (en Mbit/s)	Portée (en mètre)
11	200
5.5	300
2	400
1	500

Ces informations sur la portée sont très importantes pour bien disposer les ordinateurs portables mais aussi l'émetteur Wi-Fi correspondant aussi à la position de la caméra.

6.5.2 Interférences

Un réseau Wi-Fi fonctionne sur une bande de fréquence libre d'accès par tout matériel, ce qui rend plus difficile les transmissions sans interférence. Les principales causes d'interférences sur un réseau 802.11b ou g sont :

- ❑ Présence d'un ou plusieurs réseaux WI-FI 802.11b ou g sur le même canal ou sur un canal proche.

(Un réseau travaillant sur le canal 8 peut interférer avec un réseau du canal 6,7,9,10.)

- ❑ Présence d'un réseau Bluetooth sur cette bande de fréquence.
- ❑ Proximité de fours micro-onde en fonctionnement.

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

- ❑ Présence de tout type d'appareil communiquant sur cette bande de fréquence.

(Système de vidéo-surveillance, processeur dont la fréquence = 2.4GHZ)

Lors de la mise en place d'un réseau sans fil suivant les normes 802.11, il est donc important de minimiser le risque d'interférence, en prenant par exemple, un éloignement de trois canaux entre deux réseaux Wi-Fi.

CONCLUSION

Durant ces 5 premiers mois de stage, j'ai eu l'opportunité de reprendre en main l'ancienne version de PortaNum™, d'améliorer ses fonctionnalités, son architecture, son ergonomie...afin de la rendre la plus accessible et simple possible.

J'ai aussi réussi à faire ressortir les besoins au niveau de notre application au niveau réseau afin d'établir une version fonctionnelle pour cette rentrée scolaire.

L'application PortaNum™ possède de nombreuses perspectives puisque c'est un logiciel proposant une solution innovante, mobile, peu encombrante et utilisant du matériel grand public. Ainsi, dans les prochains mois et les prochaines années, PortaNum™ pourrait être utilisé dans de nombreux contextes (conférence, spectacle, ...) afin de faciliter l'intégration culturelle des personnes en situation de handicap. De nouveaux traitements d'images pourraient être développés et intégrés par le système de plugin permettant par exemple dans un musée d'améliorer la prise de vue pour regarder un tableau. Une modification au niveau réseau, intégration d'une liaison bidirectionnelle, permettrait de faire de PortaNum une application interactive. On pourrait imaginer deux enfants en train de travailler sur une image capturée et de collaborer afin de réaliser un projet. Ce travail collaboratif pourrait ensuite être renvoyé au professeur toujours par le biais de PortaNum.

Durant le dernier mois de stage qu'il me reste, j'espère avoir l'occasion d'expérimenter de manière plus approfondie la version réseau grâce aux nouveaux matériels. Je souhaiterais également participer à l'installation dans les écoles et ainsi avoir un retour des utilisateurs sur la nouvelle ergonomie de la version monoposte.

Durant cette période de stage, j'ai eu l'opportunité d'apprendre à travailler sur l'ensemble de la chaîne de développement d'une version d'un logiciel. J'ai pu, ainsi, travailler sur l'analyse des besoins afin de faire ressortir un cahier des charges réalisable et pour finir par le développement et un début d'expérimentation.

Le travail en équipe m'a permis de faire évoluer mes compétences ainsi que mes connaissances. J'ai aussi eu l'occasion d'effectuer des choix en tenant compte du contexte et d'une analyse préalable. Ce stage était également très valorisant d'un point de vue personnel car il m'a permis de développer un outil technologique qui puisse servir à de nombreuses personnes et qui puisse être facile d'accès.

REMERCIEMENTS

Tout d'abord je tiens à remercier le service de la Direction des Ressources Humaines France du Groupe THALES pour m'avoir donné l'opportunité de collaborer à l'élaboration du projet PortaNum. Je remercie plus particulièrement Gérard LEFRANC et Loïc GOASDUFF pour m'avoir fait confiance et m'avoir piloté durant toute la durée du stage.

Je remercie aussi tout le département de THALES Research and Technology pour leur accueil chaleureux au sein de l'entreprise. Et plus particulièrement Joseph Colineau et Hélène Soubaras, ingénieurs sur le projet PortaNum™ qui ont su m'écouter, me conseiller et m'aider tout au long de mon stage.

Merci à Aurélie Casson pour l'ordinateur qui m'a permis de réaliser mes premières expérimentations sur la norme Wi-Fi 802.11g.

Enfin, je remercie toutes les personnes rencontrées durant ce stage pour m'avoir permis de m'épanouir en rendant ce stage aussi agréable qu'enrichissant.

BIBLIOGRAPHIE

- [A001] : THALES, Groupe mondial d'électronique professionnelle
Site Internet : www.thalesgroup.com
- [A002] : Jarte, un traitement de texte très accessible
Site Internet : <http://www.jarte.com/>
- [A003] : Win32pad, un traitement de texte basique mais accessible
Site Internet : <http://www.gena01.com/prog/win32pad.shtml>
- [A004] : IEEE, comité de normalisation
Site Internet : <http://www.ieeexplore.ieee.org/Xplore>
- [A005] : Wi-Fi, groupe Wi-Fi Alliance ayant créé la norme 802.11
Site Internet : <http://www.wi-fi.org/>
- [A006] : Une description global du code correcteur d'erreur : Reed Solomon
Site Internet : http://en.wikipedia.org/wiki/Reed-Solomon_error_correction
- [A007] : Une description détaillée du code correcteur d'erreur : Reed Solomon
Site Internet : <http://lgl.epfl.ch/~barras/projects/Reed-Solomon/rs.html>
- [A008] : Indy Project, groupe de travail ayant créé la librairie de composants réseaux : Indy
Site Internet : <http://www.indyproject.org/>

Pour tout ce qui a trait au réseau Wi-Fi, j'ai utilisé :

- D. Males et G. Pujolle, *Wi-Fi par la pratique*, Eyrolles deuxième édition, 2004.
- A. Chauvin Hameau, *Wi-Fi maîtriser le réseau sans fil*, Eni Eds.
- L'article : *Video Streaming in Wireless Internet*
De Frank Fitzek, Patrick Seeling et Martin Reisslein.
- L'article : *Real-Time Caption Streaming over WiFi Network*
De D. Maniezzo_y, M. Cesanay, P. Bergamoy, M. Gerla, K. Yaoy
- L'article : *Etudes de performance des protocoles des réseaux locaux sans fil IEEE 802.11 avec introduction de la QoS*
De Racha BenAli, Hossam Afifi
- L'article : *Adaptive Forward Error Correction for Real-time Internet Video*
De Mathias Johanson

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

- L'article : *Adaptive Video Transmission Over A Single Wireless Link*
De Hao Wang, S. Venkatesan

Pour tout ce qui touche les réseaux informatiques :

- A. Tanenbaum, *Les réseaux*, Pearson Education, 2003.

Pour m'aider au développement informatique :

- *Le guide du développeur, Borland C++ Builder*

Pour tout ce qui traite l'ergonomie des logiciels :

- B.Denoëud – CNAM – Laboratoire d'ergonomie
Site Internet : www.cnam.fr

Pour tout ce qui traite de la population des personnes malvoyantes :

- Article de M. Christophe ROUSSEAU.
Doctorant en sciences de l'information – communication.
Site Internet : <http://isdm.univ-tln.fr/PDF/isdm18/54-rousseau.pdf>
- Journal Etudes et Résultats de la DREES (direction de la recherche des études, de l'évaluation et des statistiques) de juillet 2005.
Site Internet : <http://www.sante.gouv.fr/drees/etude-resultat/er416/er416.pdf>
- Article sur l'épidémiologie de la malvoyance et de la cécité.
Site Internet :
<http://www.ophtalmo.net/bv/GP/IndexGP/G/Epidemiologie/epidemiologie.htm>
- Site Internet de BlindSurfer
Site Internet : <http://www.blindsurfer.be/bsi001F.htm>

Pour tout ce qui traite l'accessibilité :

- Le site Internet du W3C : <http://www.w3.org/TR/WCAG10/>
- Le site Internet d'IBM : <http://www-306.ibm.com/able/guidelines/software>
- Article de Jérôme Ernu et Didier Garcia sur l'accessibilité du web aux personnes mal-voyantes :
Site Internet : <http://www.veblog.com/fr/2002/0527-accessibilite.html>

ANNEXES

Checklist d'IBM

IBM Software Accessibility Checklist - Version 3.5.1			
1	Keyboard access	Comments	
1.1	Provide keyboard equivalents for all actions.		
1.2	Do not interfere with keyboard accessibility features built into the operating system.		
2	Object information	Comments	
2.1	Provide a visual focus indicator that moves among interactive objects as the input focus changes. This focus indicator must be programmatically exposed to assistive technology.		
2.2	Provide semantic information about user interface objects . When an image represents a program element, the information conveyed by the image must also be available in text.		
2.3	Associate labels with controls, objects, icons and images. If an image is used to identify programmatic elements, the meaning of the image must be consistent throughout the application.		
2.4	When electronic forms are used, the form shall allow people using assistive technology to access the information, field elements and functionality required for completion and submission of the form, including all directions and cues.		
3	Sounds and multimedia	Comments	
3.1	Provide an option to display a visual cue for all audio alerts .		
3.2	Provide accessible alternatives to significant audio and video .		
3.3	Provide an option to adjust the volume .		
4	Display	Comments	
4.1	Provide text through standard system function calls or through an API (application programming interface) which supports interaction with assistive technology.		
4.2	Use color as an enhancement, not as the only way to convey information or indicate an action.		
4.3	Support system settings for high contrast for all user interface controls and client area content.		

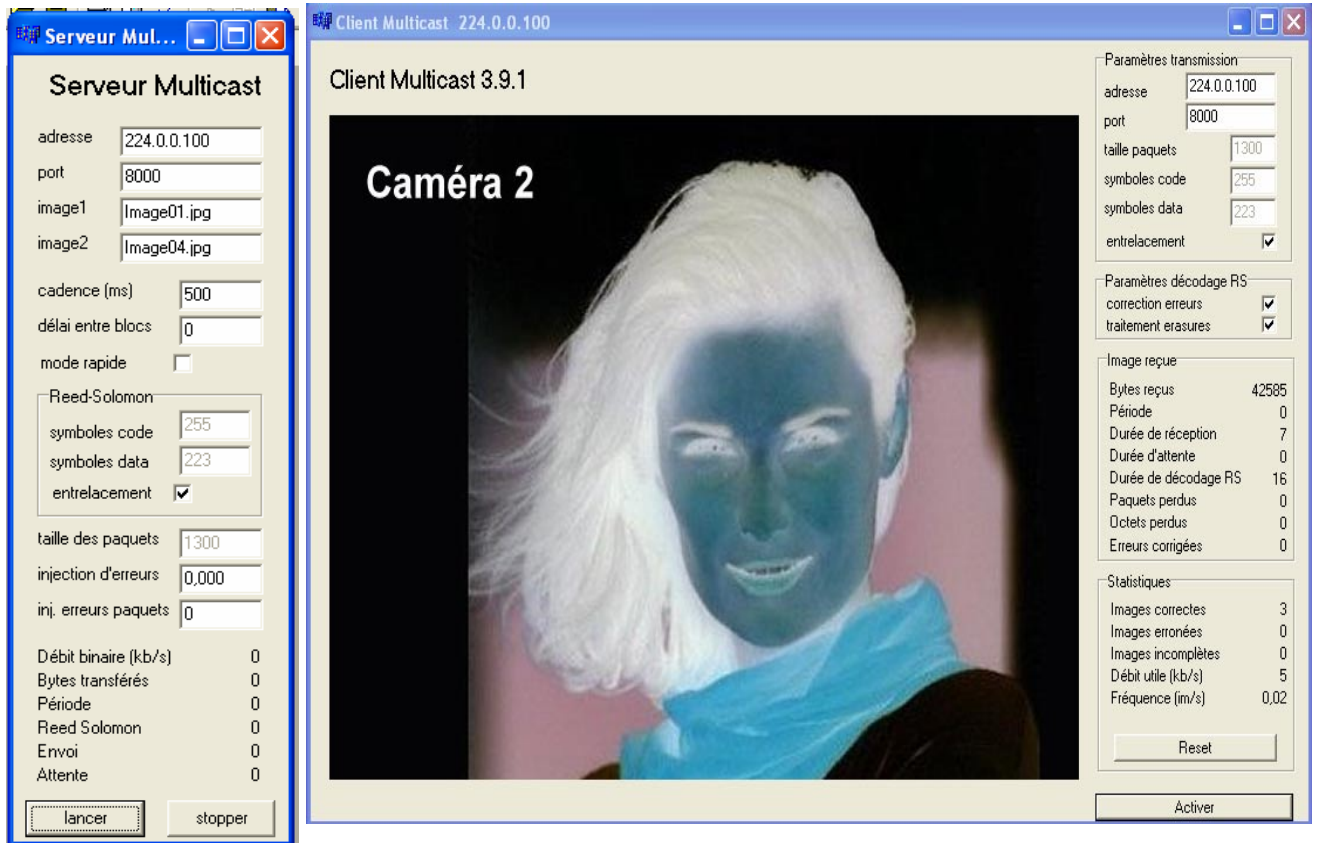
PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

4.4	When color customization is supported, provide a variety of color selections capable of producing a range of contrast levels.		
4.5	Inherit system settings for font, size, and color for all user interface controls.		
4.6	Provide an option to display animation in a non-animated presentation mode.		
5	Timing	Comments	
5.1	Provide an option to adjust the response times on timed instructions or allow the instructions to persist.		
5.2	Do not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz.		
6	Verify accessibility	Comments	
6.1	Test for accessibility using available tools.		

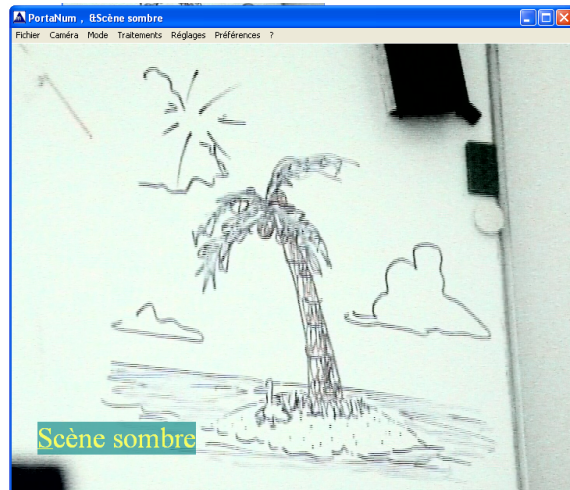
©2001, 2003 IBM Corporation

PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN

Serveur et Client du logiciel de test que nous avons créé nommé UDPTools



Capture d'écran des traitements de base



PORTANUM™ : LOGICIEL D'AIDE A LA VISION DE LOIN



Remarque : L'image capturée n'est pas la meilleure pour mettre en évidence la puissance des différents algorithmes. Cependant, on remarque que le traitement Texte manuscrit fonctionne très bien pour ce dessin tracé à la main sur un tableau.