



**HAL**  
open science

## RéoLoc : Réorganisation dynamique des caractères sur un clavier à défilement automatique ligne/colonne

Lisa Hoiry, Frédéric Rayar, Jean-Yves Antoine, Cherifa Ben Khelil, Mathieu Raynal

► **To cite this version:**

Lisa Hoiry, Frédéric Rayar, Jean-Yves Antoine, Cherifa Ben Khelil, Mathieu Raynal. RéoLoc : Réorganisation dynamique des caractères sur un clavier à défilement automatique ligne/colonne. 35e conférence internationale francophone sur l'interaction humain-machine, Mar 2024, Paris, France. hal-04450137v2

**HAL Id: hal-04450137**

**<https://hal.science/hal-04450137v2>**

Submitted on 8 Apr 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# RéoLoc : Réorganisation dynamique des caractères sur un clavier à défilement automatique ligne/colonne.

## RéoLoc: Dynamic Reorganisation of Characters on a Row/Column Scanning Keyboard.

Lisa Hoiry  
lisa.hoiry@irit.fr  
IRIT, Université de Toulouse  
Toulouse, France

Frédéric Rayar  
frederic.rayar@univ-tours.fr  
LIFAT, Université de Tours  
Tours, France

Jean-Yves Antoine  
jean-yves.antoine@univ-tours.fr  
LIFAT, Université de Tours  
Tours, France

Cherifa Ben Khelil  
cherifa.bk@gmail.com  
LIFAT, Université de Tours  
Tours, France

Mathieu Raynal  
mathieu.raynal@irit.fr  
IRIT, Université de Toulouse  
Toulouse, France

### RÉSUMÉ

Les personnes ayant des déficiences sensorimotrices rencontrent des difficultés à accéder aux dispositifs de pointage habituels et à la saisie de texte. Pour répondre à ces besoins, de multiples aides ont été développées prenant la forme de claviers virtuels. Pour les personnes ne disposant que de peu de mouvements fonctionnels, un accès par défilement sur ces claviers est privilégié. Cependant, la saisie sur ces dispositifs reste relativement lente et fatigante. Afin d'augmenter la vitesse de saisie, l'utilisation d'une prédiction de caractères permet une réorganisation contextuelle des caractères sur le clavier. Néanmoins, cette dernière n'est, en pratique, utilisée qu'avec un défilement linéaire. Cet article présente RéoLoc une approche intra ligne de la prédiction de caractères permettant son intégration avec un défilement ligne/colonne. Deux types de configurations RéoLoc ont fait l'objet d'une étude utilisateurs et ont montré leur intérêt, notamment en termes de vitesse de saisie comparé à un clavier statique.

### ABSTRACT

People with disabilities have difficulties in accessing usual pointing devices and to input text. To answer this need, several aids have been developed, including virtual keyboards. For people with limited functional mobility, scanning access is given priority. Even if a large scanning mode exists, the text input speed (TIS) is still slow. In aims to increase this TIS, some keyboards are augmented with character prediction modules. Nonetheless, this prediction is in reality used only with linear scanning. This article presents RéoLoc an intraline approach to character prediction that can be integrated with line-column scanning. Two types of RéoLoc have

been the subject of a user study and compared in term of text input speed to a static keyboard.

### CCS CONCEPTS

• **Human-centered computing** → **Accessibility systems and tools**; *Accessibility technologies*; *Accessibility design and evaluation methods*.

### MOTS CLÉS

Clavier virtuel à défilement automatique, prédiction de caractères, CAA.

### KEYWORDS

Virtual scanning keyboard, character prediction, AAC.

### ACM Reference Format:

Lisa Hoiry, Frédéric Rayar, Jean-Yves Antoine, Cherifa Ben Khelil, and Mathieu Raynal. 2024. RéoLoc : Réorganisation dynamique des caractères sur un clavier à défilement automatique ligne/colonne.. In *IHM '24 : 35<sup>e</sup> conférence Francophone sur l'Interaction Humain-Machine, March 25–29, 2024, Paris, France*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/XXXXXXXXX.XXXXXXXXXX>

## 1 INTRODUCTION

De nos jours, la communication écrite via courriel, réseaux sociaux ou encore sur les forums de discussion est l'un des principaux usages des outils numériques, plus communément appelés technologies de l'information et de la communication (TIC). La saisie de texte via les TIC (PC, smartphone, tablette, etc.) occupe une place de plus en plus importante dans notre vie quotidienne. En moins de dix ans, le smartphone est devenu un objet incontournable pour la plupart des français. En effet, selon l'édition 2022 du Baromètre du Numérique<sup>1</sup>, 87% des français de 12 ans et plus en sont équipés dont 75% d'entre eux en ont un usage quotidien.

Cependant il est essentiel de prendre conscience que cette évolution rapide ne bénéficie pas à tous les individus de manière égale. Les personnes ayant des déficiences sensorimotrices sont confrontées à des obstacles considérables lorsqu'elles tentent de tirer pleinement

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*IHM '24, March 25–29, 2024, Paris, France*

© 2024 Association for Computing Machinery.

ACM ISBN XXX-X-XXXX-XXXX-X/XX/XX...\$15.00

<https://doi.org/10.1145/XXXXXXXXX.XXXXXXXXXX>

1. <https://www.economie.gouv.fr/cge/barometre-numerique-2022>

parti de ces dispositifs technologiques. Cela souligne l'importance cruciale de réponses en termes d'accessibilité des TICs pour garantir une inclusion numérique équitable pour tous. En effet, les personnes ayant des déficiences sensorimotrices rencontrent des difficultés lorsqu'elles cherchent à utiliser des dispositifs de pointage traditionnels comme la souris et lorsqu'elles tentent de saisir du texte en utilisant un clavier physique standard. En réponse à ces défis, de multiples aides techniques ont été développées selon les capacités des personnes, mais également selon leurs habitudes de vie, les contraintes environnementales ainsi que les usages attendus de l'outil informatique [4, 12]. Nous parlons alors de communication alternative et augmentée (CAA). Pour certaines personnes, un clavier virtuel avec un dispositif de pointage alternatif suffit. Dans ce cas, le curseur peut être dirigé à l'aide d'un système de suivi oculaire (*eyetracking* [14]) ou des mouvements de la tête (*head tracking* [12, 26]) par exemple. Pour les personnes ayant des déficiences sensorimotrices plus complexes, c'est-à-dire lorsque peu de mouvements fonctionnels sont possibles, un accès par défilement sur clavier virtuel est privilégié. Sur ce type de clavier, le pointage est remplacé par un système de défilement où le curseur va parcourir le clavier touche après touche. Traditionnellement ce défilement se fait de gauche à droite et de haut en bas. Nous pouvons distinguer deux grandes catégories de défilements couramment utilisés : le défilement automatique (réalisé par un logiciel) et le défilement manuel (contrôlé par les personnes utilisatrices). Chaque principe de défilement a ses avantages et ses inconvénients et peut être personnalisé à l'aide des différents modes de défilement existants. Cependant, de nombreuses études montrent que, quel que soit le principe ou le mode de défilement utilisé, la vitesse de saisie reste relativement lente [5, 8, 10, 13, 17]. En réponse à ce constat, certains claviers sont améliorés grâce à des modules de prédiction linguistique tentant de rendre la saisie de texte plus rapide [1, 20, 23, 25]. En particulier, l'utilisation d'une prédiction de caractères permet de réorganiser dynamiquement les touches du clavier afin de limiter le nombre de défilements nécessaires pour atteindre le caractère désiré. La principale limitation de cette solution réside dans la réorganisation constante du clavier qui peut perturber la personne utilisatrice.

Dans cet article nous présentons RéoLoc (Réorganisation Locale), une approche intra ligne de la prédiction de caractères dont l'objectif est de conserver uniquement les avantages de l'utilisation de cette prédiction de manière à l'intégrer dans les claviers à défilement ligne/colonne. Cet article étudie six claviers faisant varier deux paramètres, l'intégration de la prédiction de caractères et l'organisation des lettres sur les claviers. Nous discutons des différences entre ces claviers principalement en termes de nombre de défilements moyen pour saisir un caractère, de vitesse de saisie, de temps de réaction et de préférences utilisateurs.

## 2 CLAVIERS VIRTUELS À DÉFILEMENT

Sur les claviers à défilement, le pointage est remplacé par un système de défilement où le curseur va parcourir touche après touche le clavier. Dans le cas du défilement automatique, celui-ci est réalisé par le logiciel et un contacteur permet la validation des touches. Deux contacteurs peuvent aussi être utilisés dans le cas d'un défilement manuel, i.e. contrôlé par la personne utilisatrice. Ici, l'appui

sur l'un des contacteurs permet l'avancement du défilement, tandis que l'appui sur le second permet la validation. Les scénarios de défilement sont, comme dans le premier cas, multiples [18]. Chaque principe de défilement a ses avantages et ses inconvénients : le défilement automatique permet de réduire le coût moteur (i.e. la demande physique), mais augmente la charge attentionnelle. Le système de défilement à deux contacteurs permet de réduire la charge attentionnelle, mais augmente le coût moteur. Dans notre étude, nous nous concentrons sur les claviers à défilement automatique.

### 2.1 Mode de défilement

Les claviers à défilement offrent un grand nombre de possibilités pour le mode de défilement, avec le défilement linéaire, le défilement ligne/colonne, half-and-half ou encore n-ary search.

Le défilement **linéaire** est le mode de défilement le plus simple/basique que l'on puisse avoir (Figure 1a). Le curseur va scanner chaque touche du clavier selon une progression qui se fait traditionnellement de gauche à droite puis de bas en haut, jusqu'à validation par la personne utilisatrice de la touche souhaitée par un appui sur un contacteur. Ce mode de défilement nécessite une unique validation pour saisir un caractère [24]. Cependant, le nombre de défilement pour atteindre un caractère, dans le cadre d'une organisation alphabétique, est la position à laquelle se trouve ce caractère dans l'alphabet. Par exemple, pour saisir la lettre "s", lettre fréquente dans la langue française (8%, deuxième lettre la plus fréquente en français selon le corpus[3]), l'utilisateur devra attendre 19 défilements avant de pouvoir la valider. Afin de limiter ceci, les claviers à défilement linéaire suivent, la plupart du temps,

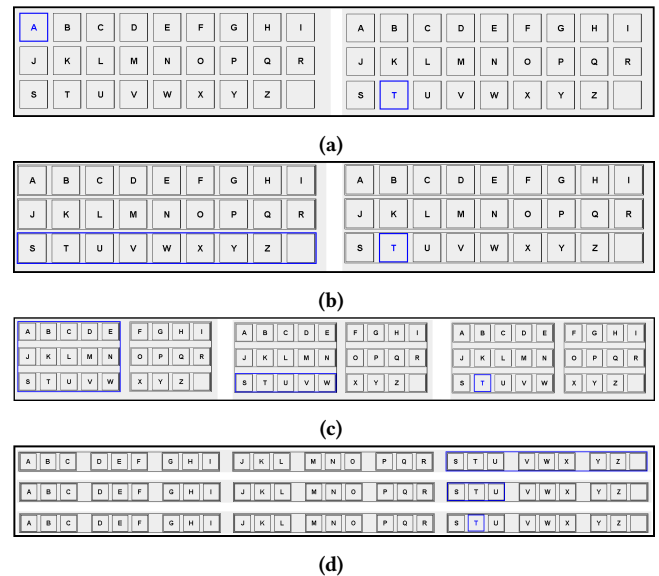


FIGURE 1 : Illustration des modes de défilements : (a) linéaire, (b) ligne/colonne, (c) half-and-half et (d) n-ary. Pour chaque mode de défilement, sont présentées les principales étapes pour sélectionner la lettre "t". Ces vues sont issues de l'outil de simulation de claviers virtuels à défilement automatique STAACS<sup>3</sup> [6]

une organisation optimisée des caractères. Cette optimisation peut se faire par une organisation où la position des caractères est établie selon la fréquence d'apparition de ceux-ci dans la langue. Dans ce cas, en français, la lettre "s" se retrouve alors en 2ème position. Une solution pour réduire encore le nombre de défilements linéaires est d'utiliser un module de prédiction de lettres afin de réorganiser dynamiquement l'ordre des caractères en fonction des saisies précédentes. Cet apport de la prédiction linguistique sera présenté en section 2.2.

Un des modes de défilement le plus utilisé est le défilement **ligne/colonne** (LC) [5, 9, 21]. Avec ce type de défilement, le curseur va dans un premier temps parcourir les lignes du clavier jusqu'à la validation, par la personne utilisatrice, de celle contenant le caractère souhaité (partie gauche de Figure 1b). Pour effectuer cette sélection, la personne devra actionner son contacteur. Après la sélection d'une ligne, le curseur scanne l'ensemble des touches qui la composent (les colonnes; partie droite de Figure 1b). Une fois encore, pour valider la touche du caractère souhaité et donc sélectionner celui-ci, la personne devra actionner son contacteur. Ici, la saisie d'un caractère nécessite deux validations. Néanmoins, dans le cas d'une disposition des lettres par ordre alphabétique sur clavier de forme de matrice diagonale, le nombre de défilement maximum passe de 27 (26 lettres de l'alphabet et l'espace) pour un défilement linéaire à 8 défilements pour un LC [11]. Même si le nombre de validations de la part de la personne utilisatrice est plus élevé qu'avec un défilement linéaire, le défilement ligne colonne permet une vitesse de saisie de texte plus rapide [24]. Notons enfin qu'en fonction des préférences de la personne utilisatrice, ce mode de défilement peut parfois être inversé et devenir un défilement colonne/ligne.

Le défilement *half-and-half* [22], quant à lui, est similaire au défilement LC. Néanmoins, le clavier est dans un premier temps divisé en deux parties, traditionnellement entre les côtés gauche et droit de celui-ci (Figure 1c). Ce mode de défilement ajoute donc une première étape consistant à sélectionner une de ces deux parties du clavier (partie gauche de Figure 1c). La suite des interactions menant à la saisie d'un caractère est identique au défilement LC. Saisir un caractère demande alors trois sélections mais le nombre de défilement peut être sensiblement réduit en fonction du caractère saisi.

Avec le *N-ary search* [17], à chaque étape, le clavier est divisé en  $N$  parties (Figure 1d). Ces parties peuvent être construites en fonction de deux stratégies. La première stratégie est la division de la section sélectionnée de sorte à avoir, autant que possible, le même nombre de lettres par partie. La seconde consiste à construire les parties en fonction de la fréquence d'apparition des lettres en équilibrant la somme des fréquences d'une partie à l'autre. Ce mode de défilement est celui qui demande le plus de sélection pour la saisie d'un caractère [17] avec un nombre de sélection égale à  $\lceil \log_N L \rceil$ , où  $L$  est la taille de l'alphabet et  $N$  le nombre de parties créées à chaque étape.

Le choix du mode de défilement peut se faire en fonction de critères comme le nombre moyen de défilements nécessaires à la saisie d'un caractère ou du nombre de sélections. Privilégier le nombre de défilements ou d'appuis dépend de la situation de chaque personne utilisatrice et est le résultat d'un processus de recommandation d'aide en ergothérapie. Dans tous les cas, il existe

de nombreuses études montrant que la saisie de texte sur un clavier virtuel statique à défilement, quel que soit le mode défilement utilisé, reste lente que ce soit en simulation [13], avec des participant-e-s témoins (i.e. personnes qui n'utilisent pas de système de CAA au quotidien) [5, 8, 17], ou encore avec des personnes utilisatrices de la population cible [10].

## 2.2 Prédiction linguistique

Afin d'améliorer la vitesse de saisie de texte, certains claviers sont perfectionnés grâce à des modules de prédiction linguistique. La prédiction de mots ou de caractères vise donc à réduire le nombre de défilements et/ou de validations nécessaires pour saisir un message, ce qui devrait théoriquement accélérer le processus de saisie de texte.

Les modèles de **prédiction de mots** opèrent en proposant une liste des mots les plus probables à saisir, en fonction soit des mots déjà insérés (prédiction), soit des caractères déjà entrés pour compléter un mot en cours de saisie (complétion). Afin d'améliorer cette liste de propositions, il est envisageable d'intégrer une dimension sémantique lors du processus de calcul des prédictions, ou encore de l'adapter en fonction du vocabulaire et de la syntaxe habituelle de l'utilisateur. Cependant, il est essentiel de souligner que la prédiction de mots s'avère souvent inefficace lorsque l'utilisateur souhaite saisir des mots qui ne font pas partie du vocabulaire prédéfini. Par conséquent, même avec l'aide de la prédiction de mots, la personne utilisatrice peut encore être contrainte de saisir un nombre conséquent de caractères pour composer son message [16, 25]. Afin de remédier à cette limitation, la prédiction de caractères apparaît comme une solution envisageable.

La **prédiction de caractères** repose sur l'analyse de la saisie en cours afin de calculer la probabilité d'entrée du prochain caractère. L'organisation des lettres sur le clavier est ensuite ajustée en fonction de la probabilité attribuée à chaque caractère, de manière à rendre plus rapidement accessible le caractère le plus probable, réduisant ainsi au minimum le nombre de défilements nécessaires à son insertion. Le calcul de prédiction de caractères s'effectue de la même manière que pour la prédiction de mots, en utilisant un arbre lexicographique [15, 19], ou encore en utilisant un modèle de langage N-gramme [25]. Les modèles N-gramme vont calculer la probabilité d'apparition d'un caractère en fonction des  $n-1$  caractères précédents déjà saisis, à partir d'un corpus de référence. Les  $n-1$  caractères servent de contexte pour la prédiction, et selon les modèles utilisés, la longueur de ce contexte peut varier. A la fin des années 90, le modèle FOCL [2] introduit cette idée en considérant uniquement la fréquence des bigrammes (2-gramme). Quelques années plus tard, SibyLetter [25] est apparu prenant en compte un contexte constitué de quatre caractères, les espaces et la ponctuation inclus (5-gramme). Une évaluation théorique de SibyLetter a montré que son utilisation permettait de réduire le nombre de défilements moyen nécessaire à la saisie d'un caractère de 33 à seulement 2,7 sur un clavier de 64 caractères à défilement linéaire.

### 3 RÉOLOC : NAVIGATION AVEC RÉORGANISATION DYNAMIQUE LOCALE

Dans le cadre du projet ANR AAC4All<sup>2</sup>, nous avons mené deux séances de focus group avec des professionnels de santé dont les patients correspondent à notre population cible (i.e. des personnes utilisant des claviers à défilement au quotidien). La première séance s'est déroulée à la Plate-Forme Nouvelles Technologies (PFNT) de l'hôpital Raymond Poincaré, avec deux ergothérapeutes et deux orthophonistes. La seconde séance a eu lieu au centre de rééducation de Kerpape, en présence de cinq ergothérapeutes, cinq orthophonistes et un utilisateur, répartis en 2 groupes.

L'objectif de ces séances était de recueillir les habitudes de saisie des personnes utilisatrices, et identifier les principaux problèmes rencontrés lors de la saisie. Les professionnels étaient invités à nous faire part de leurs retours d'expérience et nous exposer les problèmes rencontrés sur les problématiques de l'utilisation de la prédiction dans les systèmes de saisie. Pour susciter la discussion et alimenter les échanges, plusieurs propositions de claviers leur ont été faites.

Il ressort de ces séances que les personnes utilisent principalement le défilement ligne/colonne (LC). Comme nous l'avons vu dans la section précédente, la sélection d'un caractère sur un clavier à défilement LC s'effectue en deux étapes : la sélection de la ligne sur laquelle se trouve le caractère souhaité, puis la sélection de la touche. Avec cette technique, le réorganisation dynamique des caractères, en fonction de la saisie déjà effectuée n'est pas possible. En effet, avec la réorganisation dynamique des caractères, la personne n'a pas le temps de prendre connaissance de l'ensemble des caractères présents sur la première ligne avant que le curseur ne se déplace automatiquement sur la deuxième ligne. C'est pourquoi, les personnes utilisant le défilement LC conservent une disposition statique des caractères.

La réorganisation des caractères sur l'ensemble du clavier n'étant alors pas une solution envisageable, nous nous sommes intéressés à la réorganisation des caractères sur la ligne à laquelle ils appartiennent lors de la disposition de base. Pour répondre à cette problématique, nous proposons le système RéoLoc (Réorganisation Locale) qui réorganise les caractères, en fonction de leur probabilité, uniquement au sein de la ligne à laquelle ils appartiennent.

#### 3.1 Principe

Après la saisie d'un caractère, chaque ligne est réorganisée afin d'ordonner les caractères de gauche à droite, du plus probable au moins probable. Contrairement à la réorganisation dynamique que nous trouvons généralement sur les claviers à défilement linéaire[25], avec RéoLoc, les caractères ne peuvent pas changer de ligne lors de la réorganisation. A noter que ce principe est indépendant de la disposition des caractères : la réorganisation peut s'appliquer à n'importe quelle organisation de clavier, que ce soit la traditionnelle disposition AZERTY ou encore avec une disposition des caractères par ordre alphabétique ou pour des dispositions de caractères optimisés.

Dans ce contexte, pour présenter les résultats de la prédiction de caractères, nous proposons les deux stratégies que nous présentons dans les deux sous-sections suivantes.

**3.1.1 RéoLoc Globale.** Dans le cas de RéoLoc Globale (RéoLocG), toutes les lignes sont réorganisées en même temps après chaque saisie d'un caractère (Figure 2). Ainsi, sur chaque ligne, tous les caractères de la ligne sont réorganisés en fonction de leur probabilité d'utilisation. Le caractère le plus probable est en première position sur la ligne, puis le deuxième en deuxième position sur la ligne et ainsi de suite.

**3.1.2 RéoLoc Ligne.** Avec cette seconde stratégie (RéoLocL), seule la ligne sélectionnée par l'utilisateur est réorganisée, en fonction de la chaîne de caractères déjà saisie (Figure 3). A la différence de RéoLoc Globale où l'affichage est modifié directement après chaque saisie d'un caractère, avec RéoLocL, la réorganisation des caractères ne s'effectue qu'après la sélection de la ligne (Figure 3b). Ainsi, la navigation de ligne en ligne se fait toujours avec la disposition de base du clavier (Figure 3c).

#### 3.2 Implémentation

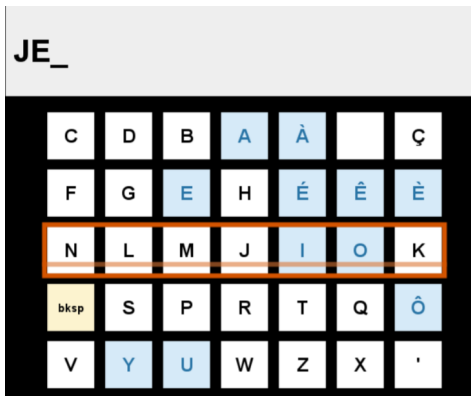
Pour pouvoir tester les différentes stratégies proposées pour RéoLoc, nous avons développé un clavier logiciel fonctionnant avec le défilement automatique du curseur en LC. Ce clavier a été développé de manière à pouvoir définir les caractères utilisables et leur disposition sur le clavier. Ainsi, RéoLoc est utilisable avec n'importe quelle disposition de caractères.

Le curseur est représenté par un rectangle de couleur brune qui englobe toute la zone sélectionnée. Par exemple, sur la Figure 3, le curseur est positionné sur la troisième ligne du clavier (Figure 3a). Dans le cas où une ligne a été sélectionnée, le curseur entoure alors la touche sélectionnée comme par exemple la touche avec le caractère 'N' sur la Figure 3b.

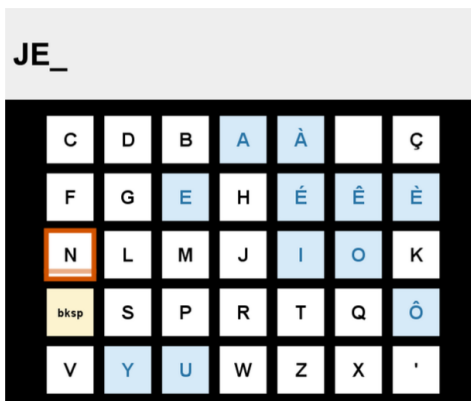
Le temps de défilement du curseur (i.e. le temps passé sur une zone avant de passer à la suivante) peut être défini séparément pour le défilement de ligne en ligne du curseur et pour celui se déplaçant de touche en touche. De plus, pour que la personne utilisatrice ait conscience du temps passé par le curseur sur une zone (ligne ou touche) une barre de progression horizontale défile de haut en bas du curseur. Lorsque la barre arrive en bas du curseur, le curseur passe à la zone suivante.

Le défilement du curseur de ligne en ligne se fait de haut en bas. Lors du démarrage du défilement, le curseur débute de la ligne la plus haute du clavier. Lorsque le curseur a atteint la ligne du bas et que la personne n'a toujours pas validé la sélection d'une ligne, le défilement reprend à la ligne du haut. Lorsque la personne sélectionne une ligne, le curseur se positionne alors sur la touche la plus à gauche de cette ligne, et défile automatiquement de gauche à droite. Si le curseur atteint la dernière touche de la ligne, et que la personne utilisatrice ne sélectionne aucune touche, le défilement repasse en mode sélection de ligne et reprend à la ligne précédemment sélectionnée. Ceci permet ainsi à la personne de sortir du défilement touche à touche sans en valider une, par exemple en cas de mauvaise sélection de ligne.

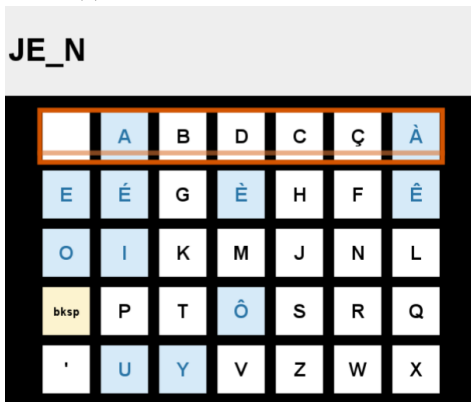
2. [www.aac4all.org](http://www.aac4all.org)



(a) Sélection de la ligne contenant le caractère souhaité



(b) Sélection du caractère souhaité

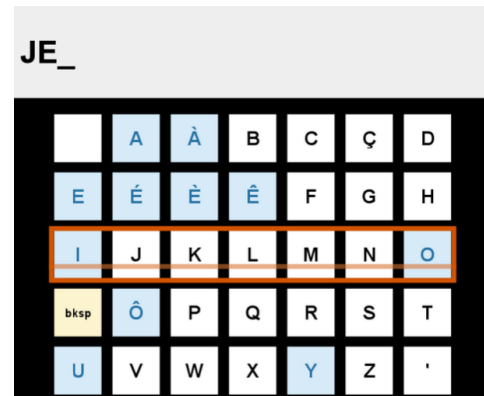


(c) Réorganisation des caractères au sein des lignes après la saisie du caractère

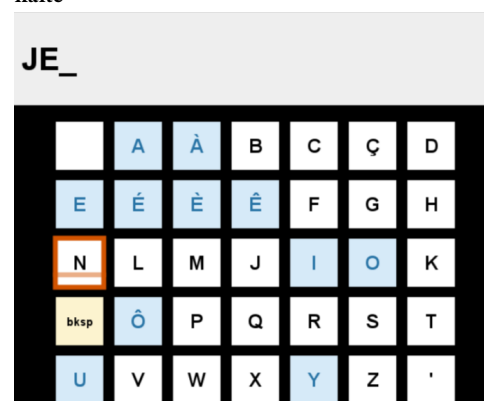
**FIGURE 2 : Illustration des étapes de saisie d'un caractère avec RéoLoc Globale**

Le module de prédiction de caractères utilisé est celui issu de l'application Life Companion<sup>3</sup>. Ce système de CAA repose sur l'utilisation d'un modèle de prédiction de caractères, SibyLetter [25], qui estime la probabilité d'occurrence d'une lettre à partir des

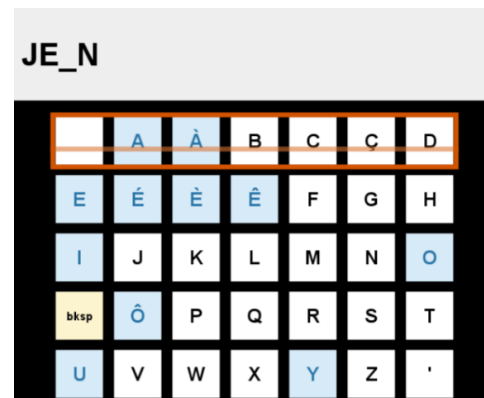
3. <https://lifecompanionaac.org/>



(a) Sélection de la ligne contenant le caractère souhaité



(b) Réorganisation de la ligne sélectionnée et validation du caractère souhaité



(c) Retour à l'organisation de base après saisie d'un caractère

**FIGURE 3 : Illustration des étapes de saisie d'un caractère avec RéoLoc Ligne**

5 caractères déjà saisis (modèle stochastique 5-gramme, espace et ponctuation étant comprises dans le contexte).

Ce clavier a été développé en respectant le formalisme de la plateforme de simulation et d'évaluation STAACS<sup>3</sup> [6]. Toutes les

informations de paramétrage du clavier (caractères affichés, disposition des caractères sur le clavier, vitesse de défilement du curseur, couleurs utilisées sur le clavier) sont définies dans un fichier XML.

### 3.3 Hypothèses

En considérant ces deux techniques de réorganisation locale des caractères, nous formulons les hypothèses suivantes : Notre première hypothèse (**H1**) est que si nous ne déplaçons pas les caractères d'une ligne à l'autre, les personnes utilisatrices ne seront pas perturbées lors de l'étape de sélection de la ligne. Par conséquent, il ne devrait pas y avoir plus d'omissions de la ligne à sélectionner avec les deux versions de RéoLoc qu'avec la version statique.

De plus, la seconde étape concernant la sélection du caractère sur la ligne s'effectue au moyen d'un défilement linéaire. Comme nous l'avons vu précédemment, la réorganisation des caractères lors d'une navigation par défilement linéaire ne perturbe pas les personnes utilisatrices lors de la saisie et permet de réduire le nombre de déplacements nécessaire au curseur pour atteindre le caractère souhaité. Notre deuxième hypothèse (**H2**) est que si nous focalisons la réorganisation de caractères uniquement sur la seconde partie de la sélection d'un caractère, cette réorganisation sera plus facilement perçue par les personnes, et le nombre de défilements à réaliser sera plus faible avec RéoLoc qu'avec le clavier statique standard.

## 4 PROTOCOLE EXPÉRIMENTAL

### 4.1 Participant-e-s

Douze personnes, recrutées dans notre laboratoire de recherche, ont participé à l'étude utilisateur, six s'auto-identifiant de genre féminin, les six autres de genre masculin. L'ensemble des participant-e-s sont âgé-e-s en moyenne de 26 ans (âge médian à 25 ans). Chaque participant-e a une expérience quotidienne de saisie de texte sur clavier virtuel à pointage (smartphone, tablette etc.) avec un clavier type AZERTY. Cependant, aucun-e des participant-e-s juge avoir une expérience avec un clavier à défilement.

### 4.2 Apparatus

L'expérience a été menée sur un ordinateur portable avec un processeur intel i7 et 16 Go de RAM. La plateforme d'évaluation est développée en Java (JDK17). La touche espace du clavier a été utilisée pour simuler un contacteur. Le temps de défilement d'un scan est fixé à 1340 ms pour le défilement des lignes et des colonnes. Ce délai a été fixé après des essais empiriques avec des personnes témoins.

Chaque version de RéoLoc (RéoLocL et RéoLocG), ainsi qu'un clavier à défilement LC dont la disposition des touches est statique (Statique) servant de clavier de référence, sont évalués. Pour notre étude, nous avons choisi d'utiliser des organisations de caractères déjà connues par les personnes utilisatrices des claviers, afin de limiter les biais d'apprentissage liés à la disposition des caractères. Pour se faire, nous avons opté pour une première organisation par ordre alphabétique (Alpha; Figure 4a), cette configuration est largement utilisée pour ce type de clavier. La seconde organisation (CV; Figure 4b) évaluée est une proposition originale de notre part validée par des professionnels de santé en lien avec notre public cible lors des séances de focus groups. Elle repose sur une séparation entre, d'une part les voyelles (accentuées ou non) ainsi que l'apostrophe

	A	E	I	O	U	Y
À	Ç	É	È	Ê	Ô	'
B	C	D	F	G	H	J
bksp	K	L	M	N	P	Q
R	S	T	V	W	X	Z

(a) Organisation CV

	A	À	B	C	Ç	D
E	É	È	Ê	F	G	H
I	J	K	L	M	N	O
bksp	Ô	P	Q	R	S	T
U	V	W	X	Y	Z	'

(b) Organisation Alpha

FIGURE 4 : Illustration des dispositions de caractères proposées dans notre étude

et le c cédille, et d'autre part, les consonnes ; ceci tout en conservant un ordre alphabétique au sein de chacun des deux groupes de caractères. Il a été décidé de tester ces deux configurations puisque (i) Alpha est la configuration la plus courante pour ce type de clavier et (ii) CV permet de regrouper les caractères en fonction de catégories (voyelles / consonnes) bien identifiées par les personnes utilisatrices, leur permettant ainsi d'anticiper leur saisie.

Chaque type de réorganisation (RéoLocG, RéoLocL et Statique) est évalué avec les organisations Alpha et CV. Ceci donne donc les six configurations de clavier suivantes : Statique\_Alpha, Statique\_CV, RéoLocL\_Alpha, RéoLocL\_CV, RéoLocG\_Alpha et RéoLocG\_CV. Pour l'implémentation des claviers, nous avons fait le choix d'intégrer les 26 lettres de l'alphabet ainsi que l'espace, disposés sur une grille de cinq lignes et sept colonnes. Par ailleurs, l'ajout des caractères accentués semblait important aux vues des retours des professionnels de santé en focus group. En effet, les personnes utilisatrices de dispositifs CAA ont à cœur d'écrire le plus précisément possible, donc sans erreur comme pourrait l'être l'omission d'un accent. Pour répondre au mieux à ces besoins, nous avons donc fait le choix d'intégrer les 7 caractères accentués les plus fréquents dans langue française<sup>4</sup> (représentant plus de 90% des usages) ainsi que l'apostrophe. De même, une touche retour, permettant la suppression du dernier caractère saisi est également présente sur notre implémentation afin de correspondre aux usages

4. Fréquences calculées à partir du corpus fourni par [3].

réels. Suite aux retours des professionnels de santé, les touches comportant des voyelles sont différenciées des autres touches par un fond bleu. De même, les touches fonctionnelles (dans notre cas seulement la touche de suppression) sont sur un fond orangé.

### 4.3 Tâche

La tâche à réaliser sur les claviers testés est une tâche de recopie. Il a été demandé aux participant-e-s de saisir un maximum de phrases par exercice, et ce, le plus précisément possible, c'est-à-dire que leur saisie doit être le plus fidèle possible à la phrase à recopier. Ces phrases sont présentées sur une ligne, et la saisie des participant-e-s apparaît sur l'éditeur de texte, une ligne en dessous (Figure 5). Les erreurs de saisie ne sont pas notifiées aux



FIGURE 5 : Interface de saisie durant un exercice de recopie

participant-e-s mais si ces dernier-e-s s'aperçoivent d'une erreur et souhaitent la corriger, iels en ont la possibilité en sélectionnant la touche « backspace » du clavier virtuel. En cas de mauvaise sélection d'une ligne, il est précisé aux participant-e-s qu'ils peuvent soit attendre la fin du défilement dans la ligne, soit saisir un mauvais caractère puis le corriger. A la fin de chaque phrase, les participant-e-s doivent appuyer sur la touche "Entrer" du clavier physique (de l'ordinateur utilisé pour l'expérimentation) afin de valider leur saisie, ce qui nettoie l'éditeur de texte et affiche la phrase suivante. Les phrases à saisir sont issues du corpus Phrases-FR-HOIRY-2023 comprenant 87 phrases [7].

### 4.4 Procédure

Chaque participant-e doit effectuer six exercices de recopie, un sur chacune des configurations suivantes : Statique\_Alpha, Statique\_CV, RéoLocL\_Alpha, RéoLocL\_CV, RéoLocG\_Alpha et RéoLocG\_CV. Chaque exercice dure 10 minutes. Une fois ces 10 minutes écoulées, l'exercice s'arrête lorsque le-la participant-e valide la phrase en cours de saisie. Une phase d'apprentissage, consistant à saisir une phrase sélectionnée aléatoirement dans le corpus Phrases-FR-HOIRY-2023 [7], précède chaque exercice. Ceci permet

aux participant-e-s de se familiariser avec la configuration du clavier à évaluer.

Les six exercices sont divisés en deux sessions, réparties sur deux demi-journées différentes afin de ne pas avoir un protocole d'étude dépassant deux heures. Ces exercices ont été répartie sur les deux sessions de manière à avoir une session pour les affichages statiques (Statique) et une autre pour les affichages dynamiques avec l'intégration de la prédiction de caractères (RéoLocL et RéoLocG). L'ordre des sessions, ainsi que des claviers au sein de celles-ci, sont contre-balançés de sorte à limiter les biais d'apprentissage. Les sessions se sont déroulées avec un intervalle maximum de cinq jours et les claviers de la première session étaient rappelés lors de la hiérarchisation de ces derniers.

A la fin de chaque exercice, il est demandé aux participant-e-s de noter la configuration sur une échelle de 0 à 100 selon chacun des deux critères suivants : la fatigue cognitive (0 si la fatigue est considérée comme très basse et 100 si elle est ressenti comme très haute) et la frustration (avec la même échelle que pour la fatigue) durant la tâche de saisie de texte. Pour éviter tout biais, les personnes répondaient sur une échelle graduée sans indication de la quantification avec comme seules indications à gauche « pas de fatigue » ou « pas de frustration », à droite « fatigue extrême » ou « frustration extrême ».

De plus, il leur est demandé de commenter leur stratégie visuelle de recherche d'un caractère autant que possible. Entre les types de réorganisation, les participant-e-s doivent classer les deux propositions d'organisation de caractères. Une fois l'ensemble des exercices de recopie effectués, les participant-e-s doivent hiérarchiser les propositions de réorganisation de clavier (Statique, RéoLocL et RéoLocG). Il est précisé aux participant-e-s de prendre en compte la sensation de rapidité de saisie ainsi que la facilité d'utilisation.

### 4.5 Données collectées

Pour chaque phrase saisie, nous enregistrons l'ensemble des actions effectuées : déplacement du curseur, activation du contacteur par la personne participante, résultats proposés par la prédiction. L'ensemble de ces traces sont stockées dans un fichier respectant un formalisme XML.

## 5 RÉSULTATS

L'ordre de présentation des claviers ne constitue pas un biais dans les données collectées lors de notre étude. En effet, nous avons effectué un test de Kruskal-Wallis et aucune des p-values n'est significative (Statique  $p = .350$ , RéoLocL  $p = .992$ , RéoLocG  $p = .758$ , Alpha-Statique  $p = .094$ , Alpha-RéoLoc  $p = .729$ , CV-Statique  $p = .731$ , CV-RéoLoc  $p = .973$ ).

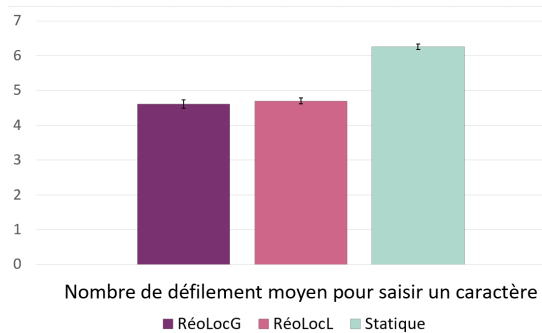
L'ensemble des données a subi un test de Shapiro-Wilk permettant d'attester ou non d'une non-homogénéité des données. Dans le cas où il n'y avait pas de raison de montrer une non-homogénéité des données, nous avons appliqué un test de Levene pour vérifier l'homogénéité des variances. Enfin, si le test de Levene ne permet pas d'attester une non-homogénéité des variances, nous avons fait une ANOVA suivie d'un test post-hoc Tukey. Dans le cas où le test de Shapiro-Wilk ne permet pas de montrer que les données sont



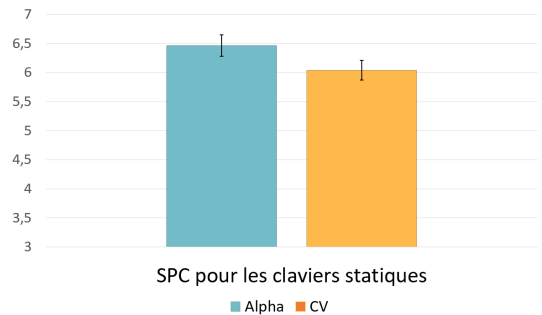
homogènes, un test de Kruskal-Wallis a été effectué suivi, si nécessaire, d'une comparaison pair-à-pair de Dwass, Steel, Critchlow et Fligner (DSCF).

### 5.1 Nombre de défilements par caractères (SPC)

Le SPC (scan-step per character [13]) permet de rendre compte du nombre de défilements effectués avant la saisie d'un caractère. Cette métrique peut être divisée en deux sous métriques :  $SPC_{ligne}$  et  $SPC_{colonne}$  [6]. Ces dernières vont respectivement rendre compte du nombre de défilement moyen sur les lignes et sur les touches (colonnes) nécessaire à la saisie d'un caractère.



(a) En fonction de la réorganisation utilisée, et indépendamment de la disposition de base des caractères



(b) En fonction de la disposition de base des caractères sur les claviers statiques

**FIGURE 6 : Nombre de défilements pour saisir un caractère (SPC) en fonction du type de disposition des lettres pour le clavier Statique ; du type de réorganisation des caractères sur le clavier. Histogramme : valeurs moyennes ; Barre verticales : intervalles de confiance à 95%.**

Selon notre étude, le nombre de SPC est réduit de 25% entre les claviers statiques et ceux de type RéoLoc ( $SPC_{moyenneReoLoc}=4,65$ ;  $SPC_{Statique}=6,25$ ; Figure 6a). La différence entre les deux types de RéoLoc est négligeable ( $SPC_{ReoLocG}=4,61$ ;  $SPC_{ReoLocL}=4,69$ . Comparaison pair-à-pair de DSCF :  $[p= .443]$ ). Sur les claviers de type Statique, nous observons une différence significative entre les deux dispositions de caractères proposées (Alpha et CV) avec un SPC plus faible pour le clavier CV ( $SPC_{Statique\_CV}=6,04$ ;  $SPC_{Statique\_Alpha}=6,46$ . Comparaison pair-à-pair de DSCF :  $p < .001$ . Figure 6b). Néanmoins, il n'y a pas de différence significative

à noter entre ces deux types de dispositions de caractères sur les réorganisations RéoLoc ( $SPC_{ReoLoc\_Alpha}=4,68$ ;  $SPC_{ReoLoc\_CV}=4,62$ . Kruskal-Wallis :  $p= .223$ ).

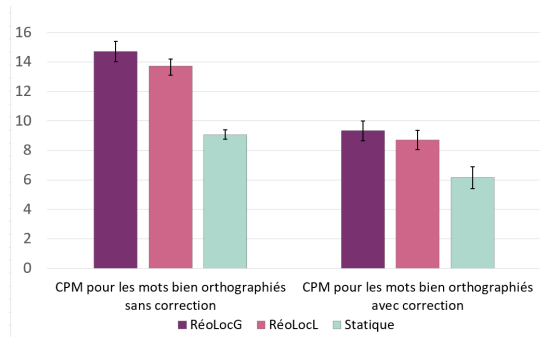
En détail, il n'existe pas de différence significative pour le  $SPC_{ligne}$  entre les trois types de réorganisation (Kruskal-Wallis :  $[p= .395]$ ). En revanche, le nombre de défilements sur les touches ( $SPC_{colonne}$ ) est réduit pour les claviers de type RéoLoc ce qui crée une différence significative sur le SPC global avec les claviers statiques ( $SPC_{ReoLocG}=4,61$ ;  $SPC_{ReoLocL}=4,69$ ;  $SPC_{Statique}=6,25$ . Comparaison pair-à-pair de DSCF RéoLocL-RéoLocG :  $p= .443$ ; RéoLocL-Statique :  $p < .001$ ; RéoLocG-Statique :  $p < .001$ . Figure 6a).

### 5.2 Vitesse de saisie : CPM

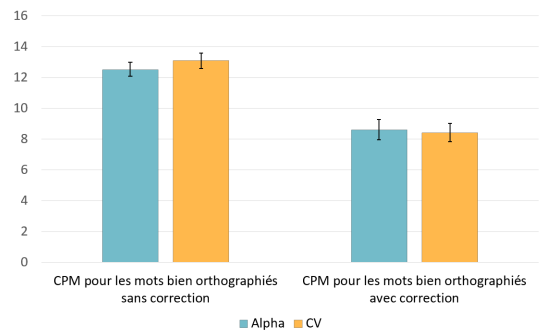
Afin d'évaluer la vitesse de saisie de texte nous avons choisi d'utiliser une mesure couramment utilisée pour l'évaluation des systèmes de saisie de texte : le nombre moyen de caractères saisis par minute (CPM). Cette mesure est calculée en comptabilisant le nombre de caractères saisis divisé par le temps mis pour les saisir, rapporté à la minute. Les caractères pris en compte dans le calcul correspondent dans notre cas à une lettre ou à l'apostrophe. Les participant.e.s étant libres de corriger ou non leur saisie, plusieurs phrases saisies comportent au moins une erreur de saisie par rapport à la phrase à saisir. Par conséquent, nous avons fait le choix de calculer la vitesse de saisie de texte sur les mots correctement saisis, c'est-à-dire ceux dont l'orthographe correspond à celle attendue, avec ou sans correction (i.e. ayant nécessité ou non l'utilisation de la touche « backspace »). A noter que les résultats sur les phrases complètes vont dans le même sens que les résultats présentés ci-dessous.

Les résultats présentés pour le SPC à savoir une diminution de 25% de cette métrique entre les claviers Statique et RéoLoc ainsi qu'un SPC plus faible pour Statique\_CV comparé à Statique\_Alpha, se reportent sur le CPM. En effet, si nous ne prenons que les claviers statiques, la différence observée pour le SPC est de nouveau présente pour le CPM sur les mots n'ayant pas nécessité de correction ( $CPM_{Statique\_Alpha}=8,84$ ;  $CPM_{Statique\_CV}=9,28$ ; Kruskal-Wallis :  $[p= .005]$ . Figure 7b). De plus, il n'existe pas de différence significative entre les claviers Alpha et CV si nous prenons en compte l'ensemble des types de réorganisation (Sans correction :  $CPM_{Alpha}=12,5$ ;  $CPM_{CV}=13,1$ ;  $[F= .178, p= .673]$ . Avec correction :  $CPM_{Alpha}=8,61$ ;  $CPM_{CV}=8,42$ ; Kruskal-Wallis :  $[p= .115]$ ).

Par ailleurs, la réduction de 25% du nombre de défilement, observée précédemment, se traduit par une diminution de la vitesse de saisie pour les claviers statiques, avec une réduction entre 30% et 40% du nombre caractères saisis par minute (Sans correction :  $CPM_{ReoLocG}=14,7$ ;  $CPM_{ReoLocL}=13,7$ ;  $CPM_{Statique}=9,06$ . Avec correction :  $CPM_{ReoLocG}=9,34$ ;  $CPM_{ReoLocL}=8,71$ ;  $CPM_{Statique}=6,15$ . Figure 7a). Malgré une différence pouvant aller jusqu'à un caractère par minute, il n'existe pas de différence significative, sur les mots avec ou sans correction, entre les deux types de RéoLoc (Avec correction :  $CPM_{ReoLocG}=14,7$ ;  $CPM_{ReoLocL}=13,7$ ;  $[F= .178, p= .185]$ . Sans correction :  $CPM_{ReoLocG}=9,34$ ;  $CPM_{ReoLocL}=8,71$ ; Kruskal-Wallis  $[p= .027]$ ). La tendance entre les types de réorganisations est conservée avec ou sans correction.



(a) En fonction de la réorganisation utilisée, et indépendamment de la disposition de base des caractères



(b) En fonction de la disposition de base des caractères, et indépendamment de la réorganisation utilisée

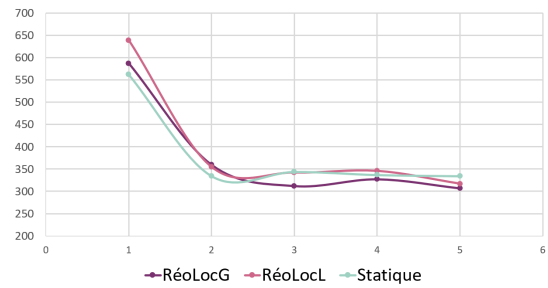
**FIGURE 7 : Nombre de caractères saisis par minute (CPM) en fonction du type de disposition des lettres; du type de réorganisation des caractères sur le clavier quelle que soit la disposition des caractères. Histogramme : valeurs moyennes; Barre verticales : intervalles de confiance à 95%.**

### 5.3 Temps de réaction

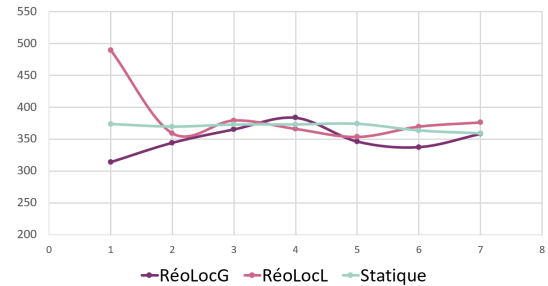
Le temps de réaction est le temps, en milliseconde, que met un-e participant-e à valider une zone une fois que le défilement a été déclenché sur cette zone. Les Figure 8a et 8b présentent ces temps respectivement pour la validation de la  $n$ -ième ligne dans un clavier et pour la validation de la  $n$ -ième touche (colonne) dans toutes les lignes, et ce pour les trois types de réorganisation étudiées.

Au vu des résultats obtenus, nous séparons leurs présentations en deux catégories : (i) le temps de réaction sur la première ligne du clavier et sur la première touche d'une ligne et (ii) le temps de réaction sur les autres lignes et sur les autres touches. En effet, les temps de réaction de cette première catégorie sont plus élevés.

**5.3.1 Première ligne et premières touches.** Sur la sélection de la première ligne du clavier, il existe une différence significative entre les trois types de réorganisation (Comparaison pair-à-pair de DSCF : RéoLocL-RéoLocG :  $p < .001$ ; RéoLocL-Statique :  $p < .001$ ; RéoLocG-Statique :  $p < .001$ ). Le temps de réaction est plus élevé pour RéoLocL puis RéoLocG et enfin Statique (RéoLocL=639 ms; RéoLocG=587 ms; Statique=562 ms. Figure 8a). Il n'existe pas de différence significative entre les deux types de disposition des caractères Alpha et CV (Alpha=608 ms; CV=593 ms. Kruskal-Wallis :  $p = .231$ ).



(a) Sélection de lignes



(b) Sélection de touches

**FIGURE 8 : Temps de réaction, en milliseconde, pour sélectionner (a) la  $n$ -ième ligne dans un clavier et (b) la  $n$ -ième touche (colonne) dans toutes les lignes, en fonction du type de réorganisation.**

Concernant la sélection de la première touche des lignes, il existe également une différence significative entre les trois types de réorganisations (Comparaison pair-à-pair de DSCF : RéoLocL-RéoLocG :  $p < .001$ ; RéoLocL-Statique :  $p < .001$ ; RéoLocG-Statique :  $p < .001$ ). Le temps de réaction est plus élevé pour RéoLocL (RéoLocL=490 ms; Statique=369 ms; RéoLocG=309 ms. Figure 8b). Il n'existe pas de différence significative entre les deux types de disposition des caractères Alpha et CV (Alpha=396 ms; CV=385 ms. Kruskal-Wallis :  $p = .141$ ).

**5.3.2 Autres lignes et autres touches.** Pour la sélection des lignes n'étant pas en première position sur le clavier, il n'y a pas de différence significative entre RéoLocL et Statique (RéoLocL=339 ms; Statique=338 ms. Kruskal-Wallis :  $p = .703$ ), qui sont les deux claviers dont l'affichage est identique au moment de la sélection de la ligne. Cependant, il existe une différence significative entre RéoLocG et les réorganisations RéoLocL et Statique (RéoLocL=339 ms; Statique=338 ms; RéoLocG=323 ms. Comparaison pair-à-pair de DSCF : RéoLocG-RéoLocL :  $p < .001$ ; RéoLocG-Statique :  $p < .001$ ), où le temps de réaction est moins élevé pour RéoLocG (Figure 8a). Il n'existe pas de différence significative entre les deux types de disposition des caractères Alpha et CV (Alpha=334 ms; CV=331 ms. Kruskal-Wallis :  $p = .082$ ).

Concernant la sélection des touches n'étant pas en première position sur les lignes, il n'existe pas de différence significative entre RéoLocG et RéoLocL (RéoLocL=361 ms; RéoLocG=352 ms. Kruskal-Wallis :  $p = .035$ ). RéoLocG et RéoLocL sont les deux types de réorganisations intégrant une prédiction de caractères et où donc

l'ordre alphabétique n'est plus respecté au sein des lignes au moment de la sélection du caractère. Il existe une différence significative entre Statique et les réorganisations RéoLoc (RéoLocL=361 ms; RéoLocG=352 ms; Statique=370 ms. Comparaison pair-à-pair de DSCF : Statique-RéoLocL :  $p < .001$ ; Statique-RéoLocG :  $p < .001$ ), où le temps de réaction est plus élevé pour Statique. (Figure 8b). Il n'existe pas de différence significative entre les deux types de disposition des caractères Alpha et CV (Alpha=363 ms; CV=360 ms. Kruskal-Wallis :  $p = .281$ ).

#### 5.4 Taux d'omission de sélection

Lorsque le temps de défilement n'est pas suffisant, une personne peut ne pas avoir le temps de sélectionner la ligne souhaitée. Pour rendre compte de ce phénomène nous avons étudié le nombre de fois où la personne n'a pas sélectionné une ligne et donc que le curseur a dû parcourir plusieurs fois le clavier. Le pourcentage d'occurrence où cette omission de sélection de ligne a eu lieu pour les claviers RéoLocG, RéoLocL et Statique est de respectivement 2, 90%, 3, 4% et 4, 43%. Il n'existe pas de différence entre RéoLocL et Statique ni entre RéoLocG et Statique (Kruskal-Wallis : RéoLocL-Statique  $p = .886$ ; RéoLocG-Statique  $p = .015$ ).

Ce taux d'omission de sélection existe également pour les touches. C'est donc le nombre de fois où la personne ne va pas sélectionner une touche dans une ligne, laisser le curseur parcourir l'ensemble de la ligne avant de la sélectionner de nouveau pour enfin valider le caractère souhaité. Le pourcentage d'occurrence où cette omission de sélection de touche a eu lieu pour les claviers RéoLocG, RéoLocL et Statique est de respectivement 1, 82%, 2, 78% et 2, 87%. Sur cette mesure, il n'existe pas de différence significative entre les types de réorganisations, ni les types de disposition (Kruskal-Wallis : Réorganisations  $p = .043$ ; Dispositions  $p = .090$ ).

#### 5.5 Taux d'erreur

Les saisies effectuées lors de l'évaluation sont de bonne qualité avec un pourcentage de mots correctement orthographiés (avec ou sans correction) supérieur à 95% (RéoLocG=98,22%; RéoLocL=97,86%; Statique=98,06%).

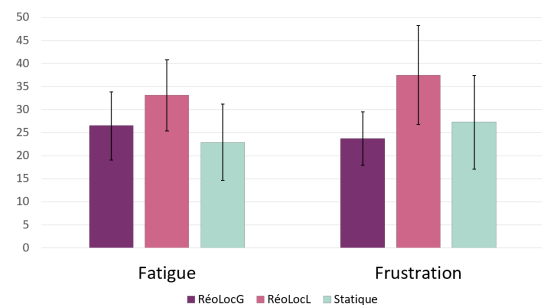
Lors de la saisie de texte, la personne peut sélectionner un caractère non souhaité. Dans ce cas, elle peut corriger cette erreur de saisie. Le taux de correction rend compte de ces erreurs. C'est le nombre de fois où la touche « *backspace* » a été sélectionnée. Le pourcentage de correction ayant eu lieu pour les claviers RéoLocG, RéoLocL et Statique est de respectivement 2, 63%, 3, 4% et 2, 29%. Il n'existe pas de différence significative entre les trois types de réorganisations (Kruskal-Wallis :  $p = .017$ ).

Les erreurs lors de la saisie de texte peuvent aussi prendre la forme de la sélection d'une ligne ne contenant pas le caractère souhaité. Pour rendre compte de ces erreurs, nous étudions le nombre de fois où la personne a sélectionné une ligne qui n'était pas celle qui contenait le caractère voulu. Le pourcentage d'occurrence où cette mauvaise sélection de ligne a eu lieu pour les claviers RéoLocG, RéoLocL et Statique est respectivement de 3, 14%, 4, 8% et 4, 67%. Il n'existe pas de différence significative entre les types de réorganisation (Kruskal-Wallis :  $p = .016$ ).

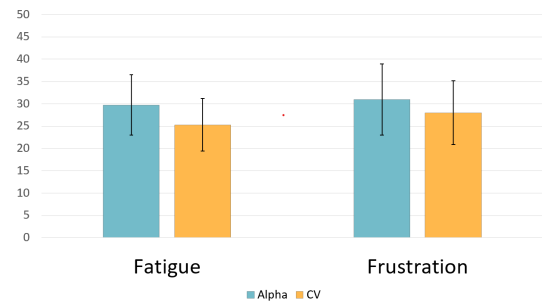
### 5.6 Préférences et retours utilisateurs

**5.6.1 RéoLocG vs RéoLocL vs Statique.** Les participant-e-s semblent plus frustré-e-s et plus fatigué-e-s lors de l'utilisation de RéoLocLigne qu'avec les autres types de réorganisations. Cependant, aucune différence significative n'est à noter entre les trois types de réorganisation (Kruskal-Wallis : Fatigue [ $p = .057$ ]; Frustration [ $p = .098$ ]. Figure 9a). Néanmoins, 10 des 12 participant-e-s ont classé RéoLocG en première position. La réorganisation RéoLocLigne n'est la préférence d'aucun-e des participant-e-s (Figure 10).

**5.6.2 Alpha vs CV.** Il n'y a pas de différence de fatigue et de frustration entre les organisations Alpha et CV (Figure 9b). Néanmoins, 10 participants sur 12 ont préféré utiliser les claviers avec une organisation CV.



(a) En fonction de la réorganisation utilisée, et indépendamment de la disposition de base des caractères

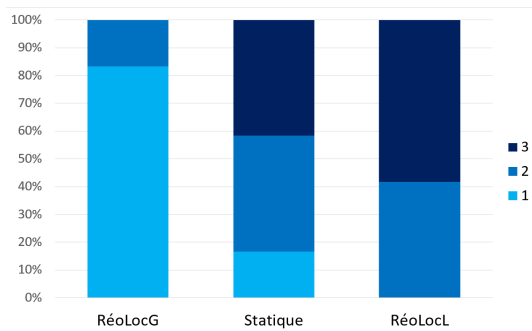


(b) En fonction de la disposition de base des caractères, et indépendamment de la réorganisation utilisée

**FIGURE 9 : Retour des personnes participantes en terme de fatigue et de frustration durant la saisie de texte en fonction (a) de la disposition des caractères (b) du type de réorganisation. Histogramme : valeurs moyennes; Barre verticales : intervalles de confiance à 95%.**

## 6 DISCUSSION

L'utilisation des résultats de la prédiction de caractères dans RéoLoc permet de réduire le  $SPC_{colonne}$  de 45,8% en pratique par rapport au clavier statique. Ceci vient donc confirmer notre hypothèse **H2**. De plus, cette réduction du nombre moyen de défilement est retranscrite par la vitesse de saisie avec un CPM plus élevé pour les claviers RéoLoc. Le gain théorique de l'intégration de la prédiction de caractère est donc conservé lors de la saisie de texte.



**FIGURE 10 : Classement des types de réorganisations de caractères par les personnes participantes à l'étude.**

D'autre part, la différence de SPC entre les deux types de RéoLoc est négligeable, ce qui est cohérent puisque ces deux types de clavier intègrent la même prédiction de caractères. Le fait que les SPC<sub>ligne</sub> des trois types de réorganisation soient identiques est également cohérent puisque nous proposons une prédiction interne aux lignes, il n'y a donc pas d'impact sur la position de la ligne contenant le caractère souhaité.

Le temps de réaction sur la sélection de la première touche des claviers RéoLocL est réellement plus élevé que pour les autres types de réorganisations. Ceci semble rendre compte des retours des personnes participantes qui sont « perturbées par le changement de l'organisation de la ligne lors de sa sélection », et trouvent cela « stressant » [P3; P4; P5; P6; P8; P9; P10]. De plus, les personnes estiment devoir « rester attentives pour ne pas anticiper la sélection de la première lettre de la ligne » car celle-ci est réorganisée et les personnes sélectionnent alors un caractère non voulu [P11; P3; P10]. Le temps de réaction pour la sélection de touches est plus élevé pour les claviers de type Statique. Ceci peut s'expliquer par les retours des personnes qui montrent que ces dernières sont moins attentives lors de la saisie sur claviers Statique que sur claviers RéoLoc. En effet, les participant-e-s estiment devoir être « plus concentré-e-s » [P3; P4; P6; P7; P8; P10; P12] que lors de la saisie sur les claviers de type RéoLoc.

Sur la sélection de ligne, le temps de réaction des participant-e-s est moins élevé sur les claviers RéoLocG que sur les claviers Statique et RéoLocL. Cette observation peut trouver un fondement dans la stratégie de recherche du caractère souhaité mise en place par les personnes. En effet, lors de la saisie de texte avec RéoLocG, elles avaient tendance à « rechercher le caractère souhaité principalement sur les deux ou trois premières colonnes » [P1; P2; P4; P5; P6; P9; P10; P11]. En adoptant cette stratégie, elles n'ont pas besoin de parcourir l'ensemble des lignes du clavier pour repérer celle contenant le caractère voulu. Ceci entraîne donc une réduction du temps de réaction sur la sélection des lignes. Cette stratégie peut également expliquer le fait que les deux versions de RéoLoc aient un nombre de tours supplémentaires moins élevé que la version statique. Cela veut dire que les participant-e-s ont omis moins souvent la sélection d'une ligne. Ceci vient donc confirmer notre hypothèse **H1** sur le fait que le nombre d'omission de sélection de ligne n'est pas supérieur avec RéoLoc par rapport au clavier statique. Notre hypothèse **H1** est également validée par les résultats sur le ressenti

des participant-e-s puisqu'il n'existe pas de différence significative sur le ressenti de fatigue et de frustration des participant-e-s entre les claviers RéoLoc et Statique.

Les personnes ayant besoin de moins de temps pour identifier le caractère souhaité et donc la ligne le contenant afin de la sélectionner, il est plus rare qu'elles n'aient pas le temps de la sélectionner. Par ailleurs, le fait que les participant-e-s mettent plus de temps à sélectionner les lignes et les colonnes en première position vérifie la recommandation selon laquelle il est préférable d'avoir un temps de défilement plus important sur le premier défilement de chaque zone.

Les saisies récupérées suite aux différents exercices de recopie sont de bonne qualité (i.e. le taux d'erreurs est inférieur à 3%), ce qui correspond à une situation de saisie standard de personnes sans troubles langagiers.

## 7 LIMITATIONS

Notre étude porte sur un clavier ne comportant qu'une partie des accents et de la ponctuation présentes en français. En effet, pour correspondre parfaitement à une saisie de texte sur un clavier physique standard, il faudrait mener une étude sur un clavier virtuel à défilement comportant tous les caractères présents sur les claviers standards (ponctuation, accents, caractères spéciaux, flèches pour le déplacement du curseur etc. [25]). De plus, la tâche de recopie ne correspond pas réellement à un usage de communication au quotidien. Il serait donc intéressant de renforcer ces premiers résultats par une étude longitudinale pendant laquelle les personnes utiliseraient ces claviers pour leurs tâches de saisie quotidiennes. Enfin, les personnes ayant participé à cette étude sont novices et n'ont aucun besoin de dispositif de CAA pour communiquer dans leur quotidien. Cette étude a donc été réalisée avec une population témoin ce qui constitue la plus grande limitation de cette étude. C'est pourquoi, les dispositifs étudiés seront testés prochainement avec des patient-e-s des centres de rééducation partenaires du projet ANR AAC4All.

## 8 CONCLUSION ET TRAVAUX FUTURS

Dans cette étude nous avons cherché à intégrer la prédiction de caractères dans un clavier à défilement automatique LC. Nous avons comparé deux types de réorganisations dynamiques des caractères : RéoLocL et RéoLocG. Ces derniers ont été confrontés à un clavier dit de référence, quant à lui statique (Statique). Nous avons également étudié l'impact de la disposition des lettres sur le clavier en comparant les dispositions Alpha et CV. Nos expérimentations ont montré que l'intégration de la prédiction de caractères interne aux lignes permet de conserver les avantages d'un clavier statique tout en améliorant la vitesse de saisie. Ce résultat est confirmé par le retour des personnes participantes qui ont en grande majorité préféré RéoLocG aux autres types de réorganisations. Il n'y pas de différence à noter en termes de vitesse de saisie entre les claviers avec une disposition Alpha et ceux avec une disposition CV. Cependant, les participant-e-s préfèrent la disposition CV nonobstant qu'ils considèrent se sentir « pressé-e-s » [P2; P3; P4; P7; P8; P11] lors de la saisie de voyelles suivies d'espaces et vice versa.

Par ailleurs, l'ensemble des observations présentées dans cet article ne s'appliquent pas forcément au public visé par ces claviers.

C'est pourquoi, une étude clinique est programmée dans les centres de rééducation partenaires du projet ANR AAC4All.

## REMERCIEMENTS

Ce travail est financé par l'Agence Nationale pour la Recherche (ANR), dans le cadre du projet AAC4All (ANR-21-CE19-0051) <https://www.aac4all.org>

## RÉFÉRENCES

- [1] Jean-Yves Antoine and Mathieu Raynal. 2022. Systèmes de saisie de texte pour les personnes avec une déficience motrice : comment les systèmes de prédiction linguistique contribuent-ils à améliorer les performances de saisie ? *1024 19* (Avril 2022), 15–26. <https://doi.org/10.48556/SIF.1024.19.15>
- [2] Tom Bellman and Scott MacKenzie. 1998. A Probabilistic Character Layout Strategy for Mobile Text Entry. In *Proceedings of Graphics Interface '98* (Vancouver, British Columbia, Canada) (GI 1998). Canadian Human-Computer Communications Society, Toronto, Ontario, Canada, 168–176. <https://doi.org/10.20380/GI1998.21>
- [3] Cherifa Ben Khelil, Frédéric Rayar, Jean-Yves Antoine, Lisa Hoiry, Mathieu Raynal, and Anaïs Halftermeyer. 2023. What you need is what you get : adapting word prediction of Augmentative and Alternative Communication aids to youth language. In *International Conference on Human-Computer Interaction*. Springer, Springer Nature Switzerland, Cham, 240–247.
- [4] N Biard, C Dumas, J Bouteille, D Pozzi, F Lofaso, and I Laffont. 2006. Apports de l'évaluation en situation de vie à partir d'une étude sur l'intérêt de la prédiction de mots auprès d'utilisateurs de synthèse vocale. In *Handicap*. 145–148.
- [5] Torsten Felzer, Ian Scott MacKenzie, Philipp Beckerle, and Stephan Rinderknecht. 2010. Qanti : a software tool for quick ambiguous non-standard text input. In *Computers Helping People with Special Needs : 12th International Conference, ICCHP 2010, Vienna, Austria, July14-16, 2010, Proceedings, Part II 12*. Springer, Springer Berlin Heidelberg, Berlin, Heidelberg, 128–135.
- [6] Lisa Hoiry, Cherifa Ben Khelil, Frédéric Rayar, Anaïs Halftermeyer, Jean-Yves Antoine, and Mathieu Raynal. 2023. STAACS3 : Simulation Tool for AAC with Single-Switch Scanning. In *17th International Conference of the Association for the Advancement of Assistive Technology in Europe (AAATE 2023)*, Vol. 306. IOS Press, 487–494.
- [7] Lisa Hoiry, Cherifa Ben Khelil, Frédéric Rayar, Anaïs Halftermeyer, Jean-Yves Antoine, and Mathieu Raynal. 2024. *Corpus de phrases pour l'évaluation de systèmes de saisie de texte en français*. Technical Report.
- [8] Farzana Jabeen and Linmi Tao. 2017. An Efficient Text Entry Model for Scanning Ambiguous Keyboard. In *9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, Vol. 1. "", 71–76. <https://doi.org/10.1109/IHMSC.2017.23>
- [9] Heidi Horstmann Koester and Simon P Levine. 1994. Learning and performance of able-bodied individuals using scanning systems with and without word prediction. *Assistive Technology* 6, 1 (1994), 42–53.
- [10] Heidi Horstmann Koester and Richard Callaghan Simpson. 2014. Method for enhancing text entry rate with single-switch scanning. *Journal of rehabilitation research and development* 51, 6 (2014), 995.
- [11] Gregory Leshner, Bryan Moulton, and D Jeffery Higginbotham. 1998. Techniques for augmenting scanning communication. *Augmentative and Alternative Communication* 14, 2 (1998), 81–101.
- [12] Edmund F LoPresti and David M Brienza. 2004. Adaptive software for head-operated computer controls. *IEEE Transactions on neural systems and rehabilitation engineering* 12, 1 (2004), 102–111.
- [13] I Scott MacKenzie. 2012. Modeling text input for single-switch scanning. In *Computers Helping People with Special Needs*. Springer Berlin Heidelberg, Berlin, Heidelberg, 423–430.
- [14] Päivi Majaranta and Andreas Bulling. 2014. Eye tracking and eye-based human-computer interaction. In *Advances in Physiological Computing*. Springer London, London, 39–65. [https://doi.org/10.1007/978-1-4471-6392-3\\_3](https://doi.org/10.1007/978-1-4471-6392-3_3)
- [15] G Ménier and Franck Poirier. 2001. Système adaptatif de prédiction de texte. *Atelier Thématique TALN 2001* (2001), 213–222.
- [16] Magali Norré and Max De Wilde. 2018. Configuration et évaluation d'un système de prédiction de mots au sein d'un logiciel de Communication Améliorée et Alternative (CAA). In *Proceedings of the Second Swiss Conference on Barrier-Free Communication (BFC 2018) : Accessibility in Educational Settings*. 71.
- [17] Ondřej Poláček, Zdeněk M'ikovec, and Pavel Slav'ik. 2012. Predictive scanning keyboard operated by hissing. In *Proceedings of the 2nd IASTED International Conference Assistive Technologies*. Citeseer, 862–9.
- [18] Ondrej Polacek, Adam J Sporka, and Pavel Slavik. 2017. Text input for motor-impaired people. *Universal Access in the Information Society* 16 (2017), 51–72.
- [19] Mathieu Raynal. 2007. Le système KEYGLASS : Système d'ajout dynamique de touches sur clavier logiciel. In *Traitement Automatique des Langues, Volume 48, Numéro 2 : Communication Assistée [Assisted communication]*, 97–121.
- [20] Mathieu Raynal and Georges Badr. 2022. Study of User Behavior When Using a List of Predicted Words. In *Computers Helping People with Special Needs*. Springer International Publishing, Cham, 331–337.
- [21] Frode Eika Sandnes. 2018. Can Automatic Abbreviation Expansion Improve the Text Entry Rates of Norwegian Text with Compound Words?. In *Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-Exclusion (Thessaloniki, Greece) (DSAI '18)*. Association for Computing Machinery, New York, NY, USA, 1–7. <https://doi.org/10.1145/3218585.3218586>
- [22] Frode Eika Sandnes, Evelyn Eika, and Fausto Orsi Medola. 2018. Multi-switch scanning keyboards : A theoretical study of simultaneous parallel scans with QWERTY layout. In *Universal Access in Human-Computer Interaction. Methods, Technologies, and Users*. Springer International Publishing, Cham, 287–298.
- [23] Keith Trnka, Debra Yarrington, John McCaw, Kathleen F McCoy, and Christopher Pennington. 2007. The effects of word prediction on communication rate for AAC. In *Human language technologies 2007 : the conference of the north american chapter of the association for computational linguistics ; companion volume, short papers*. 173–176.
- [24] Horabail Venkatagiri. 1999. Efficient keyboard layouts for sequential access in augmentative and alternative communication. *Augmentative and Alternative Communication* 15, 2 (1999), 126–134.
- [25] Tonio Wandmacher, Jean-Yves Antoine, Franck Poirier, and Jean-Paul Départe. 2008. Sibylle, an assistive communication system adapting to the context and its user. *ACM Transactions on Accessible Computing (TACCESS)* 1, 1 (2008), 1–30.
- [26] Dariusz Zapala and Bibianna Balaj. 2012. *Eye Tracking and Head Tracking – The two approaches in assistive technologies*. Hradec Králové, vCeská republika : MAGNANIMITAS, 2406–2415.