Vers un traitement automatique en soutien d'une linguistique exploratoire des Langues des Signes

Rémi DUBOT, Arturo CURIEL, Christophe COLLET
IRIT, Université de Toulouse, 118 route de Narbonne, 31062 Toulouse Cedex 9, France
prenom.nom@irit.fr

Résumé. Les langues des signes sont les langues naturelles utilisées dans les communautés sourdes. Elles ont suscitées, ces dernières années, de l'intérêt dans le domaine du traitement automatique des langues naturelles. Néanmoins, il y a un manque général de données de terrain homogènes. Notre réflexion porte sur les moyens de soutenir la maturation des modèles linguistiques avant d'entamer un large effort d'annotation. Cet article présente des pré-requis pour la réalisation d'outils s'inscrivant dans cette démarche. L'exposé est illustré avec deux outils développés pour les langues des signes : le premier utilise une logique adaptée pour la représentation de modèles phonologiques et le second utilise des grammaires formelles pour la représentation de modèles syntaxiques.

Abstract. Sign Languages (SLs) are the vernaculars of deaf communities, they have drawn interests in the natural language processing research in recent years. However, the field suffers a general lack of homogeneous information. Our reflexion is about how to support the maturation of models before starting a consequent annotation effort. In this paper, we describe the requirements of tools supporting such an approach. It is illustrated with two examples of work developed following these guidelines. The first one aims at phonological representation using logic and the second one targets supra-lexical features recognition.

Mots-clés : Langues des Signes, Formalismes de modélisation, Reconnaissance, Annotation semi-automatique .

Keywords: Sign Languages, Modeling formalisms, Recognition, semi-automatic annotation.

1 Introduction

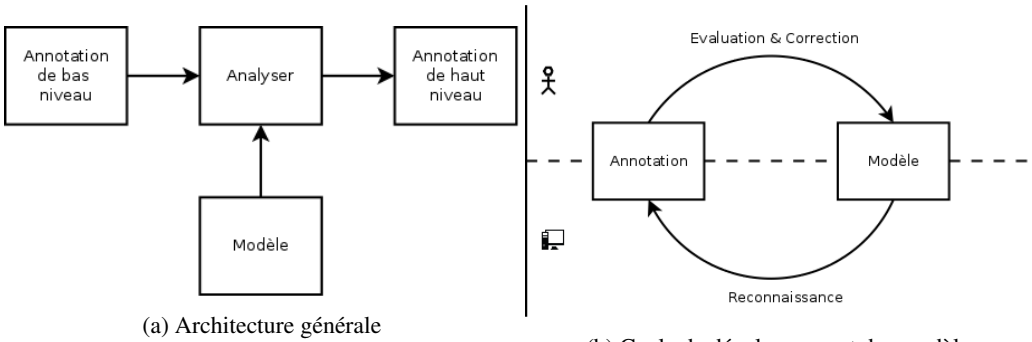
Les Langues des Signes (LS) sont les langues naturelles utilisées par les personnes sourdes comme moyen de communication principal (Stokoe, 2005). Elles se caractérisent par leur usage de l'espace en conjonction avec un ensemble de parties du corps –les articulateurs– pour véhiculer de l'information (Valli & Lucas, 2000). Bien que les mains constituent l'articulateur le plus saillant, il ne doit pas occulter le rôle essentiel de la Gestuelle Non-Manuelle (GNM) en général et de la gestuelle faciale en particulier (Cuxac, 2000). Cette multiplicité des articulateurs associée à leurs caractéristiques propres confèrent aux LS une dimension multi-linéaire non décrite pour les langues vocales. Il est maintenant bien établi que ces langues ont un pouvoir d'expression équivalent aux langues vocales.

L'intérêt de l'étude des LS dépasse largement le cadre des communautés de locuteurs. Dans le domaine linguistique, l'étude des LS contribue à l'amélioration de la compréhension des mécanismes sous-jacents de la communication humaine par la recherche de caractéristiques communes entre langues vocales et LS (Emmorey, 2001). L'intérêt croissant pour ce domaine a soulevé des questions autour de l'obtention, la comparaison et l'évaluation des données (Baker *et al.*, 2008). Les données sont toujours un point sensible en linguistique, mais l'absence de forme écrite (d'usage courant) complique encore la tâche pour les LS. Les corpus prennent la forme de documents vidéo et d'éventuelles données de capture de mouvement, accompagnés de méta-données. Une forme qui réclame des solutions de capture, de stockage et d'analyse spécifiques (Schembri *et al.*, 2014). Ces contraintes impactent lourdement les coûts de création et maintenance des corpus et par conséquence leurs tailles et leurs disponibilités.

Les problèmes posés par le support pour la création des corpus se répercutent sur l'annotation. Le processus d'annotation, bien que fondamental, est long et fastidieux (Neidle *et al.*, 2001) ; il pose des problèmes de reproductibilité des résultats, aggravé par le manque d'annotateurs (Brugman *et al.*, 2004). Même dans le meilleur des cas, la tâche n'est pas simple, les paramètres d'intérêt varient, sont nombreux et leur codage n'est pas standardisé. Tout cela motive le développement d'outils d'annotation automatique qui permettent de diminuer l'impact de ces difficultés (Dreuw *et al.*, 2010).

L'annotation automatique des LS est fortement liée à la reconnaissance de formes. Plusieurs outils ont été créés pour la détection et le suivi des articulateurs sur les images. On a d'abord vu des approches invasives utilisant des capteurs portés par le signeur comme des gants (Mehdi & Khan, 2002). Puis, avec les progrès techniques, des approches non-invasives sont apparues comme du suivi des mains et du visage dans les vidéos (Gonzalez & Collet, 2012) ou du suivi 3D avec de nouveaux capteurs (Zafrulla *et al.*, 2011). Enfin, nous sommes à un point où les solutions de reconnaissance emploient des approches très différentes, ce qui bloque une évaluation commune (Zahedi *et al.*, 2006).

Avec encore de nombreux aspects des LS mal compris et modélisés, il y a nécessité de créer des outils d'aide pour la recherche linguistique, ouverts à une démarche exploratoire. En effet, la démarche commune actuelle consiste à recourir fortement aux solutions d'apprentissage automatique développées pour les langues vocales. Outre leur adéquation sujette à débat, c'est surtout leur nécessité de large quantités de données d'apprentissage qui pose problème ici. Les démonstrations actuelles impliquent des gros efforts d'annotation spécifique pour de très petit modèles. Cela ne laisse guère de place à l'expérimentation sur les bases théoriques. En effet, les modèles appris sont fortement liés à la théorie sous-jacente à l'annotation (Zaenen, 2006). Or, étant donnée la masse de données en jeux, il n'est pas envisageable de répercuter sur toutes les annotations chaque modification de la théorie. Dans cet article, nous introduisons une nouvelle approche mettant l'accent sur l'identification du modèle (et sa formalisation) et sur l'adaptabilité à des annotations en faibles quantités et hétérogènes. Dans l'architecture adoptée, un analyseur générique utilise un modèle qui lui est donné pour compléter une annotation par la compréhension qu'il en a (cf. figure 1a). L'annotation en entré peut donc provenir d'autre outils de reconnaissance ou d'une annotation manuelle. Enfin, une évaluation de l'annotation en sortie de l'analyseur permet de corriger le modèle et donc des cycles de raffinage successifs (cf. figure 1b). Ce sont deux formalismes de représentation de modèles et deux analyseurs qui sont présentés : l'un au niveau phonologique (section 2) et l'autre au niveau syntaxique (section 3).



(a) Architecture générale

(b) Cycle de développement des modèles

FIGURE 1

2 Représentation sous-lexicale de la LS avec une logique formelle

La génération d'annotations lexicales pour les corpus est basée principalement sur la reconnaissance automatique de signes et d'autres structures. Actuellement, il existe différents outils automatiques capables de reconnaître des signes des LS, mais qui ne sont utilisables que dans des conditions très spécifiques. De ce fait, il est difficile de généraliser ces méthodes aux autres scénarios présents dans des tâches de Traitement Automatique de Langues Naturelles (TALN). Pourtant, des outils génériques sont nécessaires pour améliorer l'étendue de l'annotation des corpus et donc la quantité d'information disponible pour la reconnaissance.

Pour arriver à comprendre la LS, y compris au niveau lexical, il est nécessaire de suivre les nombreuses parties du corps du signeur (nommés *articulateurs*) qui participent à la transmission de l'information langagière. Ainsi, chacun de ces articulateurs agit comme un canal de communication à part. Pour cette raison, il est commun d'avoir des outils de capture très différents pour obtenir les données de chacun des canaux (les mains, les gestuelle non manuelle (GNM), etc). Cela montre qu'un système de reconnaissance doit être capable d'intégrer plusieurs types de sources d'informations simultanées, spécialement si son objectif est d'aller vers la reconnaissance supra-lexicale. Cependant, la diversité des représentations de signes que les systèmes existants adoptent compliquent une telle intégration. Tout cela oblige à réfléchir sur la manière de représenter les LS dans les tâches de reconnaissance, même au plus bas niveau.

Des travaux comme ceux de (Vogler & Metaxas, 2001) résolvent quelques uns de ces problèmes avec l'adoption d'une *segmentation phonologique* des signes. Ils se basent sur des modèles linguistiques comme ceux présentés par (Liddell & Johnson, 1989) et (Sandler, 1989). Dans leur système, les signes sont séparés en deux types d'intervalles temporels : intervalles de changement (*mouvements*) et intervalles statiques (*postures fixes*). En plus, les auteurs sont capables de réduire la complexité de la reconnaissance en traitant chaque canal de communication de manière individuelle, en parallèle. Pourtant, leur *framework* n'est pas facilement adaptable pour analyser des corpus sans données d'entrainement.

Par ailleurs, dans le domaine de la synthèse des LS, des auteurs comme (Losson & Vannobel, 1998) et (Filhol, 2008) ont introduit des spécifications formelles pour la description des signes. Cependant, ces formalismes sont difficiles à utiliser pour l'annotation, car ils se basent sur la construction de descriptions fondées sur des contraintes géométriques. Par conséquent, un système d'annotation basé sur ces descriptions doit évaluer des paramètres géométriques très fins, difficilement détectables de manière automatique.

2.1 La logique propositionnelle dynamique pour la langue de signes

Les auteurs (Curiel & Collet, 2013) ont défini une logique modale, la logique propositionnelle dynamique pour la LS (PDL_{SL}), pour la description formelle de la phonologie des LS. Avec cette logique, l'information phonologique de la LS peut-être représentée au travers d'énoncés atomiques. Ainsi, il est possible de modéliser des signes et d'autres structures comme des formules.

La PDL_{SL} permet également d'interpréter les corpus vidéos comme des systèmes de transition d'états (STE), où chacun des états correspond à une posture fixe et chacune des transitions correspond à un mouvement. Ces STE servent à faire de la vérification logique pour savoir si une certaine propriété des LS – ou un certain signe – existe dans une vidéo. Un exemple de ce processus est montré dans la figure 2.

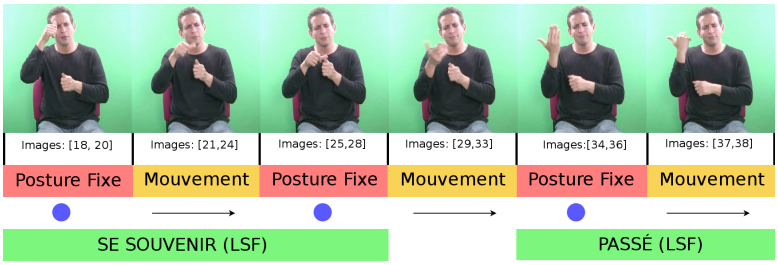


FIGURE 2 – Exemple des différentes couches présentes dans un système de reconnaissance basé sur la PDL_{SL}

L'intérêt principale d'utiliser la PDL_{SL} est d'avoir un langage formel pour décrire les actions parallèles des articulateurs. Dans ce formalisme, chacun des articulateurs peut être pris comme un agent indépendant qui agit sur une posture fixe. Cette logique permet d'exprimer l'effet des actions sur la posture de départ. Tout cela rends possible la construction de règles phonologiques de base capables d'améliorer la reconnaissance lexicale, y compris avec un manque d'information.

Le formalisme utilise le lambda calcul pour représenter de l'information incomplète. De cette manière, les informations non détectables peuvent être incluses dans les formules à travers des variables métasyntaxiques. Par exemple, les auteurs (Gonzalez & Collet, 2012) font la segmentation de postures fixes à partir de la vitesse des mains. Leur outil de suivi ne permet pas de reconnaitre les configurations des mains. Cependant, ils utilisent une mesure de proximité pour arriver à identifier si une configuration a changé. Sous le même principe, il est possible de savoir si deux mains ont la même configuration sans reconnaître exactement laquelle. La figure 3 montre une formule pour représenter cette situation. La même formule peut être utilisée aussi avec des systèmes qui arrivent à reconnaître les configurations.

$$même\_config\_deux\_mains = \lambda c.(config\_main(\texttt{droite}, c) \wedge config\_main(\texttt{gauche}, c))$$

FIGURE 3 – Formule qui représente un signe où les deux mains ont la même configuration c

2.2 Implémentation et évaluation d'un système de reconnaissance basé sur la $\mathrm{PDL_{SL}}$

Une première implémentation d'un système basé sur la $\mathrm{PDL_{SL}}$ a été développé par (Curiel & Collet, 2014). L'idée est que les utilisateurs puissent définir leurs propres bases de formules $\mathrm{PDL_{SL}}$, où chacune des formules décrit une structure lexicale d'intérêt. La solution inclut les outils nécessaires pour vérifier que les formules sont bien formées. En plus, le système intègre un solveur $\mathrm{PDL_{SL}}$, ce qui permet de trouver où chaque formule est satisfaite dans un STE. Le système génère chaque STE à partir des données de suivi d'un corpus. Chaque formule satisfaite dans un STE, représente une propriété contenue dans le corpus d'origine.

Cette version du logiciel est focalisée sur l'intégration de différentes technologies de suivi dans une même solution de reconnaissance. Il est possible de substituer les modules définis par le système avec des implémentations adaptées à d'autres outils de suivi. Cela permet de créer des modèles avec plus d'information, ce qui peut contribuer à améliorer le taux de reconnaissance. Ainsi, par exemple, le système peut être adapté pour travailler sur des données 2D ou 3D, juste en changeant de module. Pour cela, le système définit des interfaces d'implémentation. D'autre part, le même mécanisme permet d'intégrer d'autres modules de plus haut niveau pour raffiner les résultats de chaque étape, soit par des algorithmes d'apprentissage automatique ou par l'évaluation par des experts humains.

Une base de formule $\mathrm{PDL_{SL}}$ a été écrite pour permettre l'évaluation. La sortie du système a ensuite été comparée à une vérité terrain (manuellement annotée) pour produire les résultats présentés dans (Curiel & Collet, 2014).

3 Représentation de modèles syntaxiques avec des grammaires contraintes

Les linguistes montrent un intérêt croissant pour la GNM et en particulier son usage dans la syntaxe des LS. Les mécanismes de synchronisation des flux parallèles (les deux flux manuels et de nombreux non-manuels) constituent un sujet majeur dans l'étude du rôle de la GNM dans la syntaxe (Vermeerbergen *et al.*, 2007). Cette analyse a motivé le développement d'une solution pour l'annotation semi-automatique de structures syntaxiques (Dubot & Collet, 2014a).

Le formalisme de représentation de modèles présenté dérive des grammaires génératives. Il reprend l'idée introduite par Karlsson d'utiliser des contraintes pour exprimer les relations fines entre les unités (Karlsson, 1990). Il en abandonne cependant tous les détails qui rendent les travaux de Karlsson –et les travaux subséquents– spécifiques aux formes écrites des langues vocales. Le présent formalisme remplace la séquence du membre droit des règles de production par des ensembles d'unités en retirant ainsi l'information pseudo-temporelle. De tels types de grammaires sans séquences ont déjà été utilisés, au moins pour la compréhension d'images (Zhu & Mumford, 2006). L'information sur l'organisation temporelle est mise au même rang que celle pour la spatialisation ou l'articulation par exemple. De la même façon que pour les grammaires de traits, les unités sont dotées d'attributs comme, par exemple, un début et une fin dans le temps, une position dans l'espace, etc. L'organisation –temporelle et autre– est décrite par des contraintes entre les unités qui affectent les valeurs de leurs attributs. Pour restreindre le moins possible le champ des modèles représentables, la solution est compatible tant avec les grammaires lexicalisées, non-lexicalisées et mixant les deux paradigmes. Cela est rendu possible en abandonnant le concept de terminalité des symboles et en introduisant celui de détectabilité.

Le choix des grammaires génératives comme base introduit deux restrictions sur les modèles qui sont généralement considérées comme des obstacles. Premièrement, cette famille de formalismes nécessitent généralement une racine (couramment appelé axiome de la grammaire et noté S). Selon le cadre théorique dans lequel il s'ancre, un modèle peut présenter naturellement une telle racine, mais lorsqu'il s'agit d'expérimenter autour d'un phénomène syntaxique, le modèle est rarement complet et l'introduction d'une racine peut être artificielle ou même techniquement difficile. Ce problème se présente par exemple dans le cas de la modélisation par des grammaires lexicalisées (donc dont toutes les unités sont détectables) : une grammaire ne peut pas avoir une racine unique et lexicalisée, cela signifierait qu'il existe une unité gestuelle présente constamment. Ce problème est, en partie, résolu par la définition d'un ensemble de nœuds racines tel que présenté dans (Dubot & Collet, 2014a). La deuxième restriction posée par l'usage des grammaires génératives est la nécessité pour les modèles d'être complets, en particulier ceux-ci doivent descendre jusqu'au niveau des données en entrée du système de reconnaissance automatique. Par exemple, la personne voulant travailler sur les liens entre propositions d'un énoncé devra décrire dans son modèle toute la construction d'une proposition sans que ces détails soient nécessaires à son étude. La solution proposée pour ce problème est l'usage de modèles dits *boite-noires*. Ceux-ci se définissent par le fait qu'ils produisent des résultats montrant des caractéristiques globales similaires à ceux qu'un modèle classique aurait donné mais dont les détails ne sont pas interprétables. Concrètement, cela signifie que l'utilisation du modèle produit un arbre syntaxique dont la racine présente les valeurs souhaitées sur ses attributs. Par contre les nœuds sous-jacents peuvent

être inintelligibles. En reprenant l'exemple précédent, ce modèle sera capable de trouver les propositions avec les bonnes valeurs pour les attributs mais dont les constructions ne sont pas utilisables (mais aussi inutiles). Pour construire de tels modèles *boite-noires*, nous avons utilisé une forme simple de grammaires de dépendances. Les dépendances ont l'avantage d'être aisément annotables à partir des gloses [1], qui sont l'annotation la plus répandue. De plus, les résultats produits à l'aide de ces modèles sont des arbres, ce qui rend leur intégration à la solution aisée. L'intégration des dépendances au formalisme est présentée dans (Dubot & Collet, 2014b).

(a) Model (b) Result

FIGURE 4 – Exemple de modèle syntaxique

La figure 4a présente un modèle avec une seule règle de production. Il représente une description simple d'une question. Les contraintes y apparaissent comme des flèches rouges, les unités détectables comme des nœuds rouges et les unités associées à un modèle *boite-noire* en noir. La figure 4b montre un exemple de résultat de la reconnaissance automatique avec les attributs valués de façon à satisfaire les contraintes.

Parce qu'elle est conçue pour l'expérimentation d'hypothèses linguistiques, cette solution ne peut être optimisée pour une théorie linguistique en particulier. Chacun des points de divergence avec les formes traditionnelles de grammaires génératives va dans le sens de relâcher les contraintes. Ceci amène à en augmenter la complexité algorithmique. La capacité de l'analyseur à donner une solution en un temps raisonnable a été évaluée de façon empirique sur un corpus de synthèse. Les résultats montrent que l'outil (dont le code n'est pas optimisé) obtient un rappel de 20 à 80 % avec un temps de recherche de l'ordre de la seconde pour des modèles de –au moins– $\sim$ 20 unités et $\sim$ 60 règles (Dubot & Collet, 2014a). Toutefois, cette solution, en permettant de formaliser et valider des hypothèses linguistiques est en mesure d'aider à établir une base théorique suffisamment solide pour bâtir un outil d'analyse spécifique et optimisé.

4 Conclusion

Dans la situation actuelle, le manque et l'hétérogénéité des annotations est problématique. La reconnaissance automatique des LS requerra nécessairement un gros effort d'annotation à terme, en particulier pour devenir utilisable en production. Cependant, une annotation s'ancre toujours dans un cadre théorique. Le bénéfice à tirer d'une annotation de masse est donc directement tributaire de la maturité de la théorie sous-jacente. Ainsi, la qualité mais surtout la diversité des approches théoriques explorées est capitale. Pour cette raison, la position adoptée par les auteurs est celle de créer des outils dédiés à cette phase exploratoire.

Le cadre d'application visé amène à un cahier des charges particulier. Les travaux présentés dans cet article se concentrent sur la capacité à travailler avec des modèles linguistiques expérimentaux variés, et sur des données très diverses et potentiellement faibles en quantités. Par exemple, la PDL_{SL} peut être utilisée par des chercheurs en TALN pour construire et modifier rapidement des modèles lexicaux. Son usage peut aller plus loin en adaptant chaque modèle à des corpus différents, ce qui ouvre l'accès à de nouvelles sources d'information. Au niveau de l'analyse syntaxique, c'est la facilité de formalisation des modèles qui est ciblée, pour aider à la réalisation de modèles d'une partie de la langue. De cette façon, les utilisateurs sont libres de tester leurs hypothèses linguistiques.

Enfin, nous pensons que les résultats issus de la recherche sur le traitement automatique des LS sont généralisables à beaucoup de phénomènes impliquant des gestes et de la simultanéité. Par exemple, cela pourrait bénéficier à l'étude de l'intégration de la gestuelle co-verbale et de la prosodie au traitement des langues vocales.

1. En LS, les gloses sont des mots-clés représentant des signes lexicalisés.

Références

BAKER A., VAN DEN BOGAERDE B. & WOLL B. (2008). Methods and procedures in sign language acquisition studies. *A. Baker & B. Woll. Sign Language Acquisition. Amsterdam & Philadelphia : John Benjamins*, p. 1–49.

BRUGMAN H., CRASBORN O. & RUSSEL A. (2004). Collaborative annotation of sign language data with peer-to-peer technology. In *LREC, Lisbon*.

CURIEL A. & COLLET C. (2013). Sign language lexical recognition with propositional dynamic logic. In *ACL*, volume 2.

CURIEL A. & COLLET C. (2014). Implementation of an automatic sign language lexical annotation framework based on propositional dynamic logic. In *LREC Wkshp : Represent. and Process. of Sign Languages*, Reykjavic.

CUXAC C. (2000). *La langue des signes française (LSF) : les voies de l'iconocité*. Ophrys.

DREUW P., NEY H., MARTINEZ G., CRASBORN O., PIATER J., MOYA J. M. & WHEATLEY M. (2010). The SignSpeak project - bridging the gap between signers and speakers. In N. C. C. CHAIR), K. CHOUKRI & *et. al.*, Eds., *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta : European Language Resources Association (ELRA).

DUBOT R. & COLLET C. (2014a). A hybrid formalism to parse sign languages. *arXiv :1403.4467 [cs]*.

DUBOT R. & COLLET C. (2014b). Sign language gibberish for syntactic parsing evaluation. *arXiv :1403.4473 [cs]*.

EMMOREY K. (2001). *Language, cognition, and the brain : Insights from sign language research*. Psychology Press.

FILHOL M. (2008). *Modèle descriptif des signes pour un traitement automatique des langues des signes*. PhD thesis, Université Paris-sud (Paris 11).

GONZALEZ M. & COLLET C. (2012). Sign segmentation using dynamics and hand configuration for semi-automatic annotation of sign language corpora. In *Gesture and Sign Language in Human-Computer Interaction and Embodied Communication*, LNCS. Springer.

KARLSSON F. (1990). Constraint grammar as a framework for parsing running text. In *Proceedings of the 13th conference on Computational linguistics-Volume 3*, p. 168–173 : Association for Computational Linguistics.

LIDDELL S. K. & JOHNSON R. E. (1989). *American sign language : The phonological base*. Gallaudet University Press, Washington. DC.

LOSSON O. & VANNOBEL J.-M. (1998). Sign language formal description and synthesis. *International Journal of Virtual Reality*, **3**, 27—34.

MEHDI S. & KHAN Y. (2002). Sign language recognition using sensor gloves. In *ICONIP*.

NEIDLE C., SCLAROFF S. & ATHITSOS V. (2001). Signstream : A tool for linguistic and computer vision research on visual-gestural language data. *Behav. Res. Meth. Instr.*, **33**(3), 311–320.

SANDLER W. (1989). *Phonological representation of the sign : Linearity and nonlinearity in American Sign Language*, volume 32. Walter de Gruyter.

SCHEMBRI A., FENLON J., JOHNSTON T. & CORMIER K. (2014). Documentary and corpus approaches to sign language research. *The Blackwell guide to research methods in sign language studies*.

STOKOE W. C. (2005). Sign language structure : An outline of the visual communication systems of the american deaf. *Journal of Deaf Studies and Deaf Education*, **10**, 3–37.

VALLI C. & LUCAS C. (2000). *Linguistics of American Sign Language Text, 3rd Edition : An Introduction*. Gallaudet University Press.

VERMEERBERGEN M., LEESON L. & CRASBORN O. A. (2007). *Simultaneity in Signed Languages : Form and Function*. John Benjamins Publishing.

VOGLER C. & METAXAS D. (2001). A framework for recognizing the simultaneous aspects of american sign language. *Computer Vision and Image Understanding*, **81**(3), 358–384.

ZAENEN A. (2006). Mark-up barking up the wrong tree. *Comput. Linguist.*, **32**(4), 577–580.

ZAFRULLA Z., BRASHEAR H., STARNER T., HAMILTON H. & PRESTI P. (2011). American sign language recognition with the kinect. In *ICMI*.

ZAHEDI M., DREUW P., RYBACH D., DESELAERS T. & NEY H. (2006). Continuous sign language recognition-approaches from speech recognition and available data resources. In *LREC Wkshp : Represent. and Process. of Sign Languages*, p. 21–24.

ZHU S.-C. & MUMFORD D. (2006). A stochastic grammar of images. *Foundations and Trends® in Computer Graphics and Vision*, **2**(4), 259–362.