Towards Natural Language Understanding using Multimodal Deep Learning

Steven Bos

Delft University of Technology

TU Delft Delft University of Technology

Challenge the future

Towards Natural Language Understanding using Multimodal Deep Learning

THESIS

submitted in partial fulfilment of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

by

Steven Bos
born in Den Haag, The Netherlands

at the Delft University of Technology,
to be defended publicly on Tuesday March 29, 2017 at 13:00 PM.

Towards Natural Language Understanding using Multimodal Deep Learning

by Steven Bos

Abstract

This thesis describes how multimodal sensor data from a 3D sensor and microphone array can be processed with deep neural networks such that its fusion, the trained neural network, is a) more robust to noise, b) outperforms unimodal recognition and c) enhances unimodal recognition in absence of multimodal data. We built a framework for a complete workflow to experiment with multimodal sensor data ranging from recording (with Kinect 3D sensor), labeling, 3D signal processing, analysing and replaying. We also built three custom recognizers (automatic speech recognizer, 3D object recognizer and 3D gesture recognizer) to convert the raw sensor streams to decisions and feed this to the neural network using a late fusion strategy. We recorded 25 particpants performing 27 unique verbal and gestural interactions (intents) with objects and trained the neural network using a supervised strategy. We proved that the framework works by building a deep neural networks assisted speech recognizer that performs approximately 5% better with multimodal data at 20 dB SnR up to 61% better with multimodal data at -5 dB SnR while performing identical to the individual recognizer when fed a unimodal datastream. Analysis shows that performance gain in low acoustic noise is due to true fusion of classifer results while gain at high acoustic noise is due to absence of speech results as it cannot detect speech events anymore, while the gesture recognizer is not affected.

The impact of this thesis is significant for computational linguists and computer vision researchers as it describes how practical issues with (real and) real-time data can be solved such as dealing with sensor noise, GPU offloading for computational performance, 3D object and hand tracking. The speech-, object- and gesture recognizers are not state-of-the-art and the small vocabulary with 27 unique phrases and 9 objects can be considered a preliminary experiment.

The main contributions of this thesis project are a) validated multimodal fusion framework and workflow for embodied natural language understanding named MASU, b) 600GB, 2,5 hour labelled multimodal database with synchronous multi channel audio and 3D video, c) algorithm for 3D hand-object detection and tracking, d) recipe to train a deep neural network model for multimodal fusion and e) demontrate MASU in practical real-time scenario.

Faculty: Faculty of Electrical Engineering, Mathematics and Computer Science
Department: Intelligent Systems
Committee members:

Prof. M.A. Larson	TU Delft
Dr. Ir. E.A. Hendriks	TU Delft
Dr. M.J. Tax	TU Delft

This thesis is dedicated to Martien & Paula Bos and Leo & Lies Theuvenet ,
the best grandparents in the world

Contents

List of Figures vi

List of Tables vii

Preface ix

1 Introduction....1

1.1 Problem Definition 2

1.2 Research Goals 5

1.2.1 Literature research 5

1.2.2 Engineer a multimodal database 5

1.2.3 Train automatic speech, gesture and object recognizers 5

1.2.4 Train a multimodal deep neural network for semantic analysis task under noise conditions....6

1.3 Thesis Outline 6

2 Related work....7

2.1 Automatic Speech Recognition....7

2.1.1 Introduction....7

2.1.2 A brief review: from single digits to large vocabularies 9

2.2 Multimodal systems 14

2.2.1 Introduction....14

2.2.2 Multimodal fusion issues and strategies....15

2.3 Automatic Speech Understanding 17

2.3.1 Introduction....17

2.3.2 A brief review: from Machine Translation to Intelligent Assistant....20

3 Methodology....22

3.1 Introduction....22

3.2 Experiment 22

3.2.1 Setup....23

3.2.2 Task, Variables, Instruments and Measurements....24

3.2.3 Procedure 25

3.3 Data analysis 26

4 Results 30

4.1 Dataset 30
4.2 Multimodal ASU Framework 31
4.3 Automatic Speech Recognition 33
4.4 Object Recognition 35
4.5 Gesture Recognition 38
4.6 Deep Neural Network Fusion 43
4.7 Discussion 47

5 Conclusions 49

5.1 Contributions 49
5.1.1 Multimodal Automatic Speech Understanding (MASU) framework 49
5.1.2 Annotated multimodal AV Corpus for 3D scene and intent recognition 50
5.1.3 Algorithm for 3D hand-object detection and tracking 50
5.1.4 Recipe to train deep neural network for multimodal fusion 50
5.1.5 Improved Robust AV Automatic Speech Recognizer using Deep Neural Network 51
5.2 Recommendations for Future work 51

Bibliography 54

List of Figures

1-1 MASU framework model to solve the cocktail party problem..... 1
1-2 Model of communication 3
1-3 Syntax-driven Automatic Speech Understanding model. (Jurafsky & Martin, 2000) 3
1-4 Multimodal AV framework model (Galatas, Potamianos, & Makedon, 2012)..... 4

2-1 Model of communication for ASR (Jelinek, 2009)..... 8
2-2 Components of ASR systems systems (Glass J. , 2007) 8
2-3 Model of communication for ASR..... 9
2-4 Components of modern ASR system (Jurafsky & Martin, 2000)..... 10
2-5 Models of Feedforward NN, Recurrent NN and Residual NN..... 11
2-6 Model of Convolutional NN 11
2-7 Model of Early Fusion (Pradeep,2010)..... 16
2-8 Model of Interactive Fusion (Pradeep, 2010)..... 16
2-9 Model of Late Fusion (Pradeep, 2010)..... 16
2-10 Multimodal learning scenario's (Ngiam, Khosla, Kim, Nam, Lee, & Ng, 2011)..... 17
2-11 Fusion Methods (Atrey, Hossain, El Saddik, & Kankanhalli, 2010)..... 17
2-12 NLU organisation in representations (MacCartney & Potts, 2016) 18
2-13 NLU organisation in levels) (Allen, 1987) 18
2-14 NLU developement stages and timeline.. (Cambria & White, 2014)..... 18
2-15 Components of NLU system..... 19
2-16 IBM Watson's DeepQA Architecture seconds (Ferrucci, et al., 2010)..... 21

3-1 Reseach methodology 22
3-2 Render of experiment setup..... 23
3-3 Actual experiment setup 23
3-4 Kinect v1 sensor 24
3-5 Raw data 24
3-6 Experiment stages 25
3-7 Screenshot of labelling tool..... 26
3-8 WER scoring process 29
3-9 WER Improvement due to multimodal fusion Galatas et al. , 2010 29
3-10 Ideal system performance (Papandreou, Katsamanis, Pitsikalis, & Maragos, 2008) 29

4-1 Model of MASU framework 31
4-2 Implementation of MASU framework 32
4-3 Screenshot of ASR integration in UI..... 33
4-4 ASR baseline..... 33
4-5 ASR confusion matrix 20 dB SnR..... 34
4-6 ASR confusion matrix -5 dB SnR..... 34
4-7 OR pipeline..... 35
4-8..4-13 OR algorithm phases 35-36
4-14 OR baseline 37
4-15 OR confusion matrix..... 37
4-16 GR pipeline
4-17..4-20 GR algorithm phases..... 38-39
4-21 Screenshot of GR integration in UI..... 39
4-22 Gesture model 40
4-23 GR Baseline..... 40
4-24 GR confusion matrix 41

4-25 OR+GR baseline 41
4-26 OR+GR confusion matrix 41
4-27 DNN architecture..... 43
4-28 DNN training plot..... 44
4-29 SR+OR+GR+DNN performance..... 44
4-30 SR+OR+GR+DNN with baselines overview..... 45
4-31 SR+OR+GR+DNN confusion matrix 20 dB SnR..... 45
4-32 SR+OR+GR+DNN confusion matrix -5 dB SnR..... 46
4-33 SR+OR+GR+DNN performance with ideal detector..... 46

List of Tables

1-1 Actions and goals of our research.....5

2-1 ASR Characteristics.....7
2-2 Six levels of coorperation.....14
2-3 ASU Characteristics.....19

3-1 Overview of 27 classified intents.....26

4-1 ASR baseline.....33
4-2 OR baseline.....36
4-3 OR precision/recall.....36
4-4 GR baseline.....39
4-5 OR+GR baseline.....41
4-6 SR+GR+OR+DNN performance.....44
4-7 SR+GR+OR+DNN performance with ideal detector.....46

Preface

This thesis is the very proof that tenacity always pays off. I started my Master of Science graduation project at the TU Delft in 2011, after being inspired by dr. Pascal Wiggers during his course Real-Time AI and Speech Recognition. He stated that despite all effort, the problem of natural language understanding is still a largely unanswered question. As a student Multimedia Knowledge Engineering and later Computer Science, the crossroads of AI, Semantics and Theory of Mind being materialized in a robot or agent program is wildly enticing. The result was opening Pandora's Box, the cover illustration of my 100+ pages research assignment on prior Natural Language Understanding research and the biological and computational components involved. It was a solid base to build the computational framework and a first proof-of-concept found in this work: a better speech recognizer by letting it see. Altough the end result, in addition to about 8000 lines of code, is also the cover illustration painted by my grandmother, it can also be considered as a step closer towards automatic speech understanding (but still worlds apart).

I am truly grateful for the inspirational courses at the TU Delft and Leiden University that resulted in a firm scientific engineering toolbox: Prof. Rothkrantz (Neural Networks), dr. Lew (Multimedia Information Retrieval), Prof. Erik Jansen (3D Computer Graphics and VR) and Prof. Jonker (Artificial Intelligence Techniques).

The supervision of my project was *nearly* impossible and I would like to express a huge thanks to dr. Pascal Wiggers, dr. Emile Hendriks and dr. Judith Redi. This thesis would not be possible without the great support and freedom given by Prof. Henk Scholten, dr. Marianne Linde, drs. Steven Fruijtier and drs. Sanne Hettinga at Geodan Research. Furthermore, I would like to thank drs. Dagmar Stadler and dr. Anita Coetzee for the opportunity and great time to work with students and teaching staff as a student assistant at the TU Delft.

The final big thanks are for my great friends Dave and Rob who had to miss many beers and snowboard holidays due to this work. Also Brian, Kafung, Chun, Farid, Bas, Joeri and the many friends at GKV, thanks for all the wonderful insights and years of friendship. Of course without my brother, sister, parents, aunt Raphael and the whole Bos and Theuvenet family encouraging me to no end also helped quite a bit :). Especially Daan Bos and Paul Theuvenet who inspired me early on to pursuit a career in science and engineering. My greatest appreciation goes to my guiding light in the past 12 years: Jessica. Thanks for believing in me and bearing the weight!

Steven Bos, Den Haag, March 2017

1

Introduction

"*Meaning is use and structure emerges from use*". These two aphorisms come from (Tomasello, 2008) after relentlessly studying young children and primates. *Use* involves actions, so to understand this world we need to sense and interact with it. The capabilities for sensing and interacting are bound to our body, thus limiting what we can perceive, process and achieve – and thus understand. This philosophical position is the *embodied theory of mind*, the human body defines how it understands the world. Through sensing and interacting it learns new knowledge that is always grounded in bodily experience. For example, we cannot understand the meaning of the word "yellow" without seeing the color and form a direct association between the language concept and the sensory-motor experience of the body. These experiences contain what the body sensed at that time– what it heard, saw, felt, thought. (Roy, 2003) proposed four stages of *embodied learning*, showing that each concept we can think of, no matter its abstraction, must eventually be placed in one or more "boxes", whom in turn are always grounded in sensory-motor experiences.

Most trained models and machines do not follow the principles of embodied learning and are *ungrounded*. This affects its generalization to other tasks and domains as it will not be able to crossover since there is no common basis or shared building blocks. Thus the amount of knowledge that can be captured and the way to communicate is fixed. The recent popularity of *machine learning* and especially *deep learning* (LeCun, Bengio, & Hinton, 2015) show that deep learning systems can generalize surprisingly good with enough data and without changing the system architecture (eg. in Automatic Speech Recognition (Sutskever, Vinyals, & Le, 2014) and for various Computer Vision tasks (Oquab, Bottou, & Laptev, 2014). However, as the majority of these systems are ungrounded, they require human domain experts to preprocess the input data and interpret the results. Moreover, no current system can generalize their knowledge to use in completely unrelated domains and for other tasks. In the field of Artificial Intelligence, creating a system that can generalize knowledge over unrelated domains is an ambitious goal stated as *general artificial intelligence*, more or less a simulation of the human brain. Embodied learning in combination with deep learning algorithms is an exciting approach towards this goal.

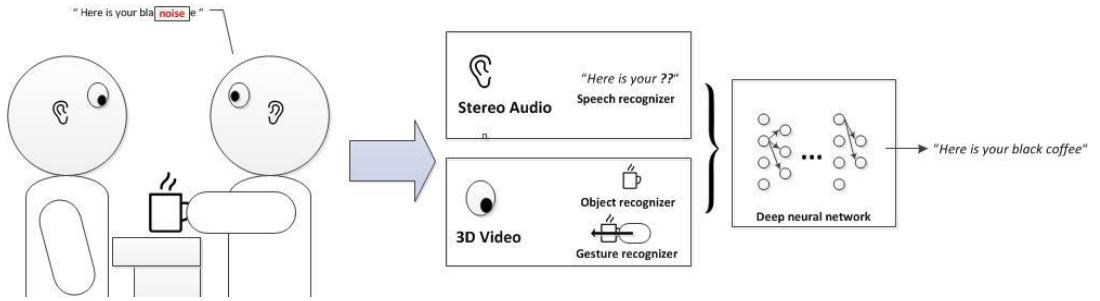


Fig 1-1 *Our multimodal Automatic Speech Understanding(ASU) framework approach to solve the cocktail party problem using embodied learning (fusing what it hears and sees)*

This thesis dives into a subset of embodied learning, grounded language learning (Mavridis & Roy, 2006; Reckman, Orkin, & Roy, 2010; Gorniak & Roy, 2003). It introduces a new computational framework (see Fig 1-1) to process multimodal (stereo audio and 3D video) data in real time and demonstrates its application in a practical and challenging use case – the cocktail party problem. In this problem, background noise affects speech signals thus hindering the cognitive ability to understand speech. Humans are able to cancel out sources of noise and infer speech signal based on visual context and discourse. Most automatic speech recognizers (ASR) have no such abilities and recognition performance of audio only ASRs degrade fast in noisy conditions. We are not the first to use multimodal data fusion or use 3D data to improve ASRs in noisy conditions eg. (Mroueh, Marcharet, & Goel, 2015) but to our knowledge are the first to do so using audio, object and gestures cues captured from stereo audio and 3D video in combination with deep learning for multimodal fusion.

Advances in grounded language learning have many implications for society in the short and long term. Research by (Ruan, Wobbrock, Liou, Ng, & Landay, 2016) show speech input outperforms type input three times over. In the short term chatbots like Microsoft's Cortana, Google Now or Apple's Siri that are integrated in smartphones could be improved by using both the microphone and camera('s) to analyse both the surrounding as well as the user. Such modern personal assistants can profile your behavioral patterns, assist and fill in your knowledge gaps. Ideally, it can not only answer, but also explain how it got to that answer (Knight, 2016), how confident the assistant is in its inference and how reliable its sources are. In the longer term, cars such as Tesla's could improve their autonomous driving experience through better scene prediction. By simply subscribing to various weather and social mediastreams a car can be aware of and peek into the future - it can learn correlations between a scene description such as "people walking on the highway" or a photo geotagged in Amsterdam with snow and a probable future scene with appropriate car behavior.

1.1 Problem Definition

This thesis investigates if we can improve an off-the-shelf automatic speech recognizer (ASR) by recognizing speech, objects and gestures in two modalities: stereo audio and 3D color vision. This leads to the following research question:

"Can multimodal fusion of stereo audio and 3D video improve ASR performance "

Automatic Speech Recognition

Automatic speech recognizers have become a prevalent technology to be found in virtually all modern smartphones and modern web browsers. It is embedded in personal assistants or chatbots mentioned before and form the core of the system. Without proper speech-to-text conversion (the goal and scope of ASRs), no question or intent from the user can be distilled and no intelligent response can be given - the principle of *garbage in, garbage out*. The applications of ASR go well beyond personal assistants. For humans, one-way communcation through sound already develops during pregnancy. Having a robust ASR enables an interaction paradigm with any device or system that is natural, convenient and expressive. Compared to other input paradigm such as gaze, touch, keyboard and mouse input, speech relieves the eyes and hands for other tasks while providing input at huge speeds and expressiveness as seen in human-human communication.

Automatic speech recognition is considered a computationally hard problem for decades, with minor improvements in recognition rate each year. With the recent advent of deep neural networks major breakthroughs have been achieved. Microsoft Research (Xiong, et al.,

2016) has recently shown an ASR that performed on par with the *gold standard*, with an error rate of less than 5%. Although they show that we can create ASR's at the same performance level as humans in a range of practical scenarios, the problem is still largely unsolved. Variation in speech patterns (e.g. accents and child directed speech which is slower and high pitched), unknown words like uncommon names and vocabulary size in general, speaker independence (no training phase), spontaneous speech (in contrast with well pronounced, isolated speech) and most important any adverse condition such as noise, degrade the ASR performance.

Channel noise

For this thesis we focus on one adverse condition that affects ASR performance: background noise , strictly defined as *channel noise* (see Fig. 1-2). Channel noise is interesting because it is inevitable noise in most use cases. This thesis uses a Gaussian white noise source at various Signal to Noise Ratios (SnR), to validate our system. ASR performance under noisy conditions is measured for an audio only baseline and compared against the proposed multimodal ASU using the industry standard Word Error Rate (WER) metric.

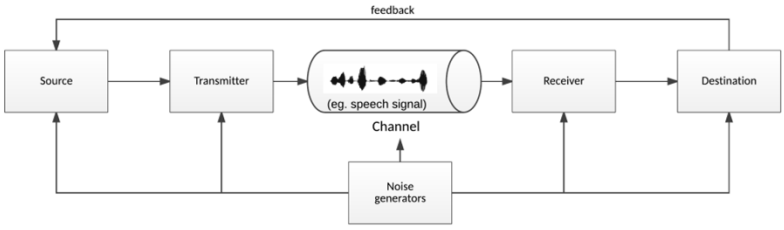


Fig 1-2 *Basic model of communication with noise effects possible in every stage*

Embodied Automatic Speech Understanding with Computer Vision

The scope of ASR is limited to converting raw speech to strings of words, which in limited domains and certain conditions is solved. The next frontier (Zweig, 2016) is *automatic speech understanding* (ASU) or Natural Language Understanding (NLU), which extends ASR's scope from words to a meaning representation[1]. Typically this process is modelled as a syntax-driven semantic analysis pipeline (Jurafsky & Martin, 2000) as in Fig 1.3, which assumes compositionality (the meaning of a sentence can be composed from the meaning of its parts).

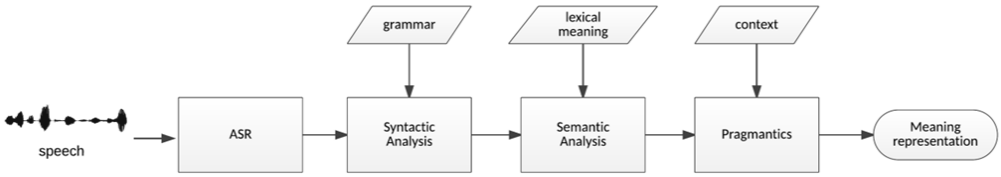


Fig 1-3 *A syntax-driven Automatic Speech Understanding (ASU) pipeline*

While the computational linguistic approach in Fig 1-3 is worthwhile exploring, it goes against the principle of embodiment discussed earlier. To understand meaning requires a wholistic approach rather than a focussed approach. One approach would be to introduce more (temporally aligned) data sources such that strongly correlated audio-visual events can be learned. (Roy, 2003) empirically shows that children do this similarly. They learn and correctly use words for the first time by being exposed to them in specific contexts.

[1] with the definition of meaning discussed in the introduction. For a more elaborate discussion on this topic see [Bos, 2012]

Using more and temporally aligned data also addresses the confusability problem in statistical ASR systems ie. when multiple outcomes share similar probabilities other modalities can be checked for finding evidence that supports these outcomes. Research by (Feldman J. , 2006) shows that humans rely on vision more than other senses. Or, as professor Fei Fei Li mentions, "if we want our machines to think we need to teach them to see[2]", making computer vision (CV) the modality of choice next to audio for this thesis.

Advances in sensor technology and machine learning solve many traditionally hard CV problems. Cheap 3D sensors such as the Microsoft Kinect, Leap Motion and Scanse Sweep LIDAR allow milimeter resolution depth sensing while a specific type of deep neural networks, convolutional neural networks allow for breaktroughs in image classifcation, gesture recognition and other CV tasks.

Multimodal fusion

With isolated ASR and CV systems independent streams of outcomes are produced. To reach a single outcome or conclusion, multimodal fusion is required. This is a non-trivial task as two or more independent systems can generate opposing outcomes. With a large number of independent systems simple averaging could work due to the Law of Large Numbers, however if one or more sources hold more truth or are less sensitive for noise, their outcomes get absorbed. A typical multimodal architecture is presented by (Galatas, Potamianos, & Makedon, 2012) and is adapted for this thesis in Fig. 1-4.

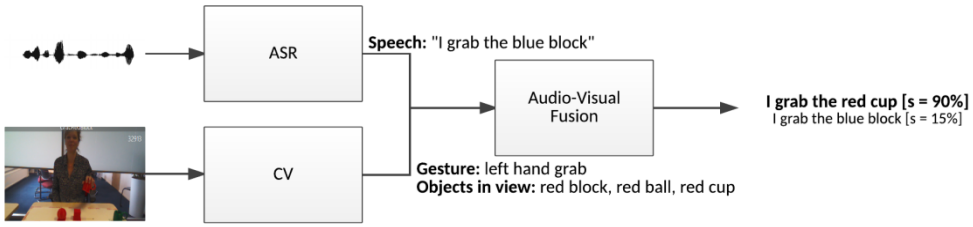


Fig 1-4. *General multimodal framework fusing audio and video streams into an ordered list of outcomes*

To summarize, a working hypothesis (H_0) can be formulated from our research question:

"Fusing classified objects and gestures from a 3D camera with speech have a beneficial effect on the WER performance of an Automatic Speech Recognizer under increasing Gaussian white noise condition from 20 dB up to -5 dB SnR compared to speech only."

The architecture, results, discussion and conclusion will appeal to researchers and engineers in the field of robotics, computional linguistics, computer vision and human computer interaction.

[2] http://www.wired.com/2015/04/fei-fei-li-want-machines-think-need-teach

1.2 Research Goals

We have broken down our research question in the following actions and research goals, formulated in table 1.1:

Table 1-1 *Actions and goals of our research*

Action	Goal
1.1 Literature research in ASR, MM systems, DNN	Learn about the state-of-the-art
1.2 Engineer 3D video recorder and replayer	Create a multimodal database
1.3 Train ASR for English spontaneous speech	Create an ASR baseline
1.4 Train gesture & object recognizer with 3D data	Create a Computer Vision (CV) baseline
1.5 Train multimodal semantic analyser using DNN	Prove that ASR+CV is better than ASR in isolation (H_0)
1.6 Engineer framework for real-time processing	Create real-time system w/ multi core & gpu offloading

1.2.1 Literature research

Automatic Speech Recognition

A great body of literature has been written on ASR. For this section we aimed for a brief introduction about the principles of ASR, its models, mathematical foundation and how deep neural networks have changed ASR research completely.

Multimodal Systems

The problem of multimodal fusion itself is not trivial. For this section we aimed for a brief introduction on existing multimodal systems fusing audio and video and how multiple approaches exist to do raw feature fusion, decision fusion or hybrid fusion.

Automatic Speech Understanding

Key part of this thesis is using deep neural networks to perform semantic analysis on the multimodal data, in other words trying to understand speech. For this section we aimed to give a brief introduction on the principles of ASU, its models and to describe a brief history of existing ASU systems.

1.2.2 Engineer a multimodal database

As this is novel research, we developed our own multimodal database to train our semantic analyser. Existing multimodal databases, even those recorded with RGBD sensors such as the Kinect often aim on gesture and object data, see (Firman, 2016) for an overview. No dataset was found with RGBD data that also recorded the correlating audio nor recorded a real human actor interacting with objects using speech and gestures. Recording this data was challenging as the default Kinect V1 studio software did not provide software for simultaneous audio and rgbd-video recording and replaying, thus custom software for both recording and replaying was developed. The collected dataset is available for the research community.

1.2.3 Train automatic speech, gesture and object recognizers

Our research questions required a framework where speech, gesture and object recognizers operate on synchronised audio and video data. No such (open source) framework was found in the research community. For the individual recognizers, we used one pre-trained model (Microsoft Speech Platform v11 English acoustic model trained for the Kinect sensor) while all

other models were designed and trained for this thesis. As 3D image recognition is still novel, a significant part of this research will focus on designing a 3D image preprocessor for object recognition. The framework is designed in such a manner that any recognizer can be replaced with another one to compare performances.

1.2.4 Train a multimodal deep neural network for semantic analysis task under noise conditions

The most important contribution of this thesis is centered on training a multimodal deep neural network and therefore proving our hypothesis to be correct. We achieved it by engineering an audio noise generator, experiment with various deep neural network architectures and perform hyperparameter optimization to discover a performing trained model.

1.3 Thesis Outline

Chapter 2 summerizes related work by other researchers, covering multimodal audiovisual ASR and ASU systems, multimodal fusion strategies and key concepts of deep neural networks including convolutional and recurrent neural networks.

Next, in chapter 3 our research methodology is discussed with exact descriptions on how our experiment was executed. Performance metrics for our baseline and solution, our measured variables, the collected data and our analysis pipeline is all discussed.

For both engineers and scientists the most interesting part will be chapter 4, where we present and discuss both our implementation and results.

Finally, in chapter 5 conclusion, we will revisit our working hypothesis and discuss our contributions and propose exciting directions for future work.

2

Related work

This chapter describes the state-of-the-art in ASR research with a brief history on the pre-statistical modelling era and current era with Deep Neural Networks. A list of related work on multimodal systems and multimodal fusion in general is discussed next. Prior work in deep learning relevant for this thesis (convolutional neural networks) is covered next. Finally, a brief review of ASU research is provided, connecting above topics into one application.

2.1 Automatic Speech Recognition

2.1.1 Introduction

Before we dive in a brief history on speech recognition, it is best to start with some context and semantics. Our working definition for ASR:

"*ASR is the process of automatically converting spoken language (speech) to language symbols (eg. words, sentence)*"

This process is found in literature under various names such as speech recognition (SR) and speech-to-text (STT). ASR systems are useful in isolation or as part of a larger system. In isolation, ASR systems are great for data entry such as dictation applications or using your voice for command and control applications (e.g. piloting a swarm of robots). As part of a larger system, ASR systems are more than just a natural interface between humans and computer devices. For instance, when part of an ASU system (eg. an intelligent agent or chatbot) the output of ASR systems are processed beyond language symbols to a richer representation. This representation can be anything, such as human intents, a series of fitting responses or, as in this thesis, a corrected or more complete string of language symbols.

Characteristics for ASR systems

There are many variables that make the process of converting speech-to-text difficult. They include phonological variations like dialect, individual differences such as the anatomy of the voice box or socio-linguistic factors or real-world issues like made-up or new words (see (Glass J. , 2007) for more variables).

With many variables, many characteristics can be identified to classify ASR systems ((Jurafsky & Martin, 2000) (Glass J. , 2007):

Table 2-1 *ASR Characteristics*

Vocabulary size	small (<20 words) to large (>50000)

Perplexity	low (<10 possibilities for each word) or high (>247)
Speaking mode	isolated word or continuous speech
Speaking style	read speech or spontanuous speech
Enrollment speaker	dependent or speaker independent
Adverse conditions	distance, noise, type of microphone

Summed up, the holy grail are large vocabulary, speaker independent, spontaneous and continuous ASR systems that perform well in noisy (low signal-to-noise ratio) environments. When discussing ASR systems in this thesis we mean large vocabulary continuous speech recognizers (LVCSR), the term used often in speech recognition literature, unless stated otherwise.

Modelling ASR systems

The automatic speech recognition problem is traditionally modelled as Shannon's Information Theory problem using the noisy channel model, an insight from Jelinek and Bahl in 1975 (Jelinek, 2009) see figure 2-1. In short, the model assumes that noise deforms the clean source signal and the receiver has to estimate the source signal from the noisy signal. This model simplifies the communication model of 1.2 by treating all noise as channel noise.

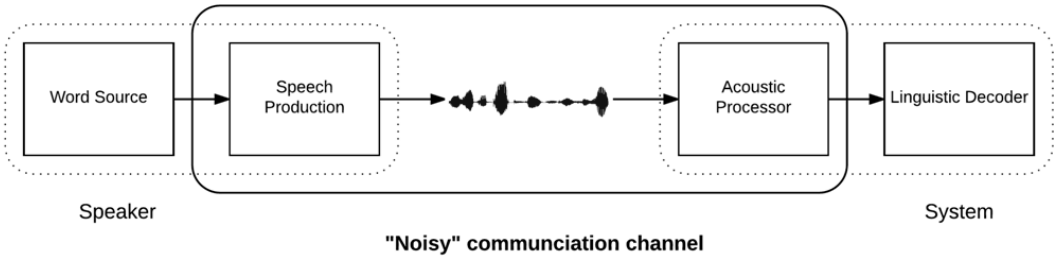


Fig 2-1. The original noisy channel model modelled for statistical ASR (Jelinek, 2009)

Implementation of the noisy channel model require solutions to three problems (Jurafsky & Martin, 2000), a metric for the best match between noisy speech and speech and an algorithm to efficiently search in all possible words. Thirdly, the implementation requires one or more knowledge representations of the speech signal such that it can be used for computations. Modern ASR systems (Glass J. , 2007) exploit constraints by modelling them (eg. the word "xab" doesn't exist in English) and use these models to assist the search algorithm. Typical constraints are phonemes[3] in acoustic models, pronounced words in lexical models and sentences in language models, see figure 2-2.

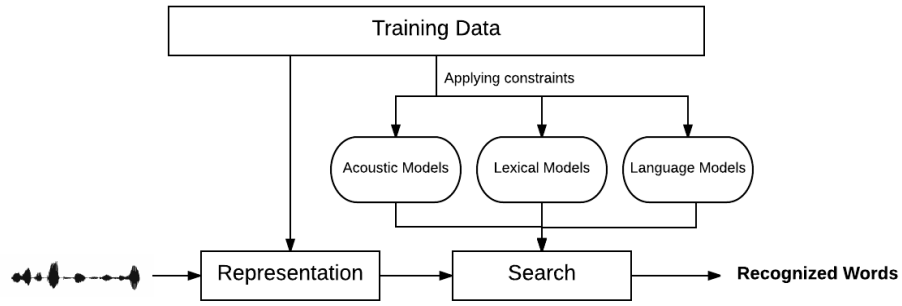


Fig 2-2. *Main components of ASR systems* (Glass J. , 2007)

[3] a phoneme is a perceptually distinct unit of sound in a language , eg. the letter x in English is phoneme /ks/ or /gz/

2.1.2 A brief review: from single digits to large vocabularies

Template-matching ASR systems

The first ASR systems used pattern matching techniques to recognize isolated digits in the speech signal of a single speaker (Davis, Biddulph, & Balashek, 1952). Two decaded later (Itakura, 1975) (Martin, 1970) and (White & Neely, 1975) created ASR systems capable of recognizing 30-200 words of a single speaker with 99% accuracy in continuous speech. These systems only performed well under ideal circumstances - no background noise, a known (trained) speaker, clear articulation. The work of (Reddy, 1976), (Juang & Rabiner, 2004), (Jurafsky & Martin, 2000) and (Jelinek, 2009) provides a great technical overview of the period from 1950-1976. They remark the dominance of the *dynamic time warping*, handwritten rule-based and template-matching techniques in that era: a deterministic, white-box pattern recognition approach.

Probabilistic ASR systems

Already in the late '50 statistics were introducted to build a phoneme recognizer (Fry & Denes, 1959). However, the wide spread adoption of probability theory in speech recognition was discouraged until the late '70 (Liberman, 2010) (Jurafsky & Martin, 2000) in part due to the influential linguist Chomsky's arguments that probabilities can generate grammatically correct English sentences that do not occur in the English discourse. It was not untill 1976 that the work of (Baker, 1975) and (Jelinek, 1976) paved the way for modern speech recognizers (Huang & Deng, 2010) based on probabilistic noisy channel implementation (see figure 2-3) and Hidden Markov Models HMM) for acoustic models and intermediate knowledge representation for decoding.

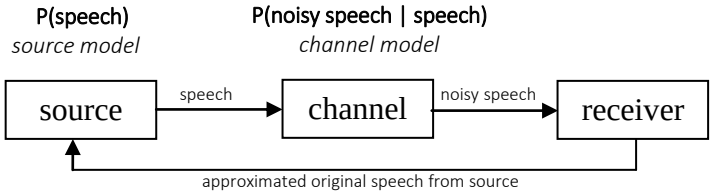


Fig 2-3. *Noisy channel model with probabilistic model of source and channel*

This probablistic noisy channel model inf Fig 2.3 is mathematically formalized as

$$\widehat{W} = \arg\max_{W \in L} P(W|O) \qquad (1)$$

$\widehat{W}$ = the approximated spoken sentence
W = actual spoken sentence
L = the language containing all possible spoken sentences
O = observed (noisy) spoken sentence
$P(W|O)$ = probability of seeing the correct spoken sentence given that we observe some noisy speech

It translates to "*what is the most likely sentence (consisting of words W) out of all possible sentences in the language L given some observed acoustic input O*".

With the help of Bayes' Rule we can decompose the P(W|O) term to

$$\widehat{W} = \arg\max_{W \in L} \frac{P(O|W)P(W)}{P(O)} \qquad (2)$$

And finally to (3) as the term P(O) is the same for each candidate in spoken sentence W

$$\widehat{W} = \arg\max_{W \in L} P(O|W)P(W) \qquad (3)$$

Interestingly, now the part P(W) can be considered a language model, the probability of seeing a spoken sentence in the language L. The other part, P(O|W) can be considered an acoustic model, the probability of observing noisy spoken speech given some spoken sentence. A typical implementation is depicted in figure 2-4 (Jurafsky & Martin, 2000).

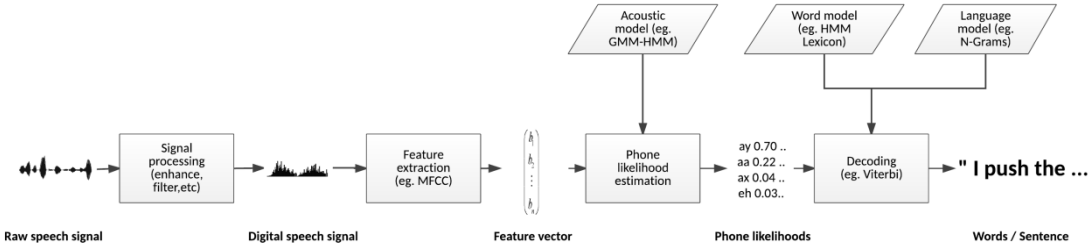


Fig 2-4 *Typical statistical architecture of a modern Automatic Speech Recognizer (ASR)*

Deep neural network ASR systems

ASR implementations such as in figure 2-4 have been using HMM's and N-grams for their acoustic and language model for the last fourty years (Huang, Baker, & Reddy, 2014). However, since 2010 with the popularization (eg. (Schmidhuber, 2015), (LeCun, Bengio, & Hinton, 2015) and application of deep neural networks for speech recognition (eg. (Seide, Li, & Yu, 2011)) major breakthroughs were achieved. The current state-of-the-art ASR systems of Microsoft Research (Xiong, et al., 2016) and IBM (Saon, Sercu, Rennie, & Kuo, 2016) perform around the gold standard (human performance) of about 4.1% - 9.6% (Xiong, et al., 2016). They use the same noisy channel model approach except substituted the HMM and N-gram techniques for deep neural networks such as Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN). The decoding part is still done with the classical HMM intermediate knowledge representation only generalized as Weighted Finite State Transducers (WFST, see (Mohri, Pereira, & Riley, 2008)). The state-of-the-art ASR systems use an upgraded decoding algorithm of the classical (Viterbi, 1967) to a parallel version (Mendis, Droppo, Maleki, Mytkowicz, & Zweig, 2016).

The term deep learning in relation to neural networks, often refers to amount of hidden layers of a neural network (NN) topology. The exact amount of layers that define NN as deep (as opposed to shallow) is an area of discussion for DL researchers (Schmidhuber, 2015). Schmidhuber, being one of the DL pioneers, defines any NN with 10 or more layers as "very deep". Yann Le Cun, another DL pioneer defines "deep" if it the network has more than one stage of non-linear feature transformation (Cun, 2016)) . Surprisingly, the field of deep learning in neural networks date back to approximately the invention of the first artifical neural networks, the Perceptron (Rosenblatt, 1957). Already in 1965 (Ivakhnenko & Lapa, 1965) published about a deep (feedforward) multilayer perceptron, but it took until 2000 until the label *deep learning* was explicitly introduced in the context of hidden layers in Neural Networks (Aizenberg, Aizenberg, & Vandewalle, 2000).

Progress in the field of deep neural networks is tightly linked to ASR systems and for each new NN one or more ASR implementations has been devised and been published. We briefly present three types of Deep Neural Nets that influenced or is in use by the current state-of-the-art ASR sytems.

Multi Layer Perceptrons (MLP)
With the (re)introduction of hidden layers and backpropgation to train these layers (Rumelhart, Hinton, & Williams, 1986) MLP's overcame the limitations stated by Minksy and Papert in 1969[4], and were able to approximate any given continuous function. These MLP's, essentially deep neural networks, achieved moderate success in acoustic modelling (Bourlard & Morgan, 1994), (Tebelskis, 1995) for ASR systems. Its potential compared to HMM approaches is greater with less assumptions (Tebelskis, 1995).

MLP network are a class of feedforward neural networks (fig 2-5) that map input data to outputs, with outputs being useful for tasks like regression analysis, classification, decision support, motor control etc. The core idea of training feedforward neural networks (with backpropagation) is controlled adjustments of the weights in the nodes such that input data result in desired output data. This is done by comparing outputs with *desired* outputs and distribute the error (the difference) back over the network such that the weights of the nodes are adjusted. Various forms of error distribution can be chosen (per node, layer or whole networks) as well as parameters for adjusting the weights (learning rate, forgetting rate, momentum, etc). Feedforward networks are acyclyc graphs and train well on static data such as images. One of the more popular feedforward networks, and proven to be very suitable for speech recognition, are convolutional neural networks (CNN) (fig. 2-6). The principle of convoluting two input signals is great for transforming the speech signal to images (Dai, 2016) and finding local correlations. These networks are also used in the winning solutions for both image recognition benchmarks (He, Zhang, Ren, & Sun, 2016) and is used for both acoustic and language modelling (Arisoy, Sainath, Kingsbury, & Ramabhadran, 2012) in the current state-of-the-art ASR system (Xiong, et al., 2016).

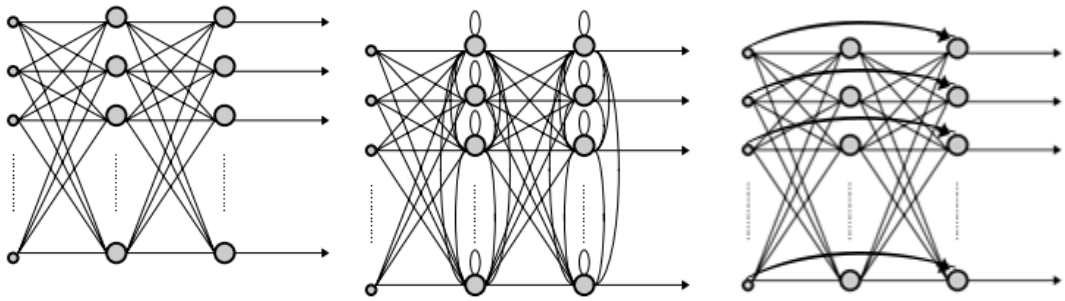


Fig 2-5 *Feedforward NN* *Recurrent NN*

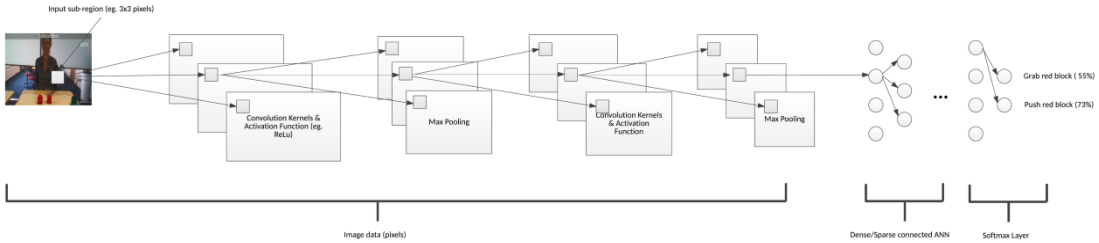


Fig 2-6 *Convolutional neural networks, a deep learning architecture for image recognition*
Recurrent Neural Networks (RNN)

[4] **The AI Winter**, see: https://en.wikipedia.org/wiki/History_of_artificial_intelligence#The_first_AI_winter_1974.E2.80.931980

To model sequential or *temporal* (time-series) and parallel data, information needs to be learned that go beyond a static state in a feedforward network. By knowing context, or the preceeding and expected values of a state, the network can calculate the most likely path and outcome. RNN's are the deepest type of NN that can be trained, as this network is a cyclic graph and can process inputs of arbitrary lengths.

The cyclic property makes it useful for sequential data, but at the same time are difficult to train, suffering from the so called *fundamental deep learning problem* or *vanishing or exploding gradients problem* (Hochreiter, 1991). This also applies to DNN like the MLP's discussed earlier. Various solutions to this problem have been invented to address the problem like rectified linear unit (ReLU) instead of sigmoids or hyperbolic tangent activation functions (Gloror, Bordes, & Bengio, 2011), residual networks (Veit, Wilber, & Belongie, 2016), multi-level hierarchy (Schmidhuber, 1992), Batch normalization (Dai, 2016) and LSTM networks (Hochreiter & Schmidhuber, 1997).

In relation to ASR systems, the DNN approach based on RNN's outperformed the probabilistic approach based on Gaussian Mixture Model(GMM)-HMM's for the first time in 2013 (Graves, Mohamed, & Hinton, 2013). LSTM networks, a variant of RNN that use memory cells for long range interdependencies, are currently used in all state-of-the-art ASR sytems eg. (Xiong, et al., 2016) and (Saon, Sercu, Rennie, & Kuo, 2016). Most recently, RNN Grammars (Dyer, Kuncoro, Ballesteros, & Smith, 2016) show that RNN's are also useful for learning state-of-the-art language models.

AutoEncoders

A major difficulty in machine learning is the preprocessing of data to speed up or improve learning. This is often a time consuming (costly), manual labor. Tasks involve labelling, cleaning and transforming (feature extraction) of data. NN's that are trained using human preprocessed data are classified as supervised learning NN's. The holy grail of machine learning is the other extreme, unsupervised learning, where NN detect patterns such as anomalities in data without any human intervention (al-Dosari, 2016), or document similarity (Salakhutdinov & Hinton, 2009).

Between the extremes is semi-supervised learning where NN's are trained on largely unlabeled data (and minimal labeled data) or with human intervention during the training process. One type of neural network that excel in unsupervised and semi-supervised learning are *autoencoders* (Heck, Konig, Kemal Sönmez, & Weintraub, 2000) and (Hinton, Osindero, & Teh, 2006). These networks try to extract abstract features automatically (Bengio, Courville, & Vincent, 2014) and represent the input data in a lower dimension. This is done by imposing constaints on the networks such as fewer neurons in the hidden layers than in the input layers, on the activation function like in sparse autoencoders (Ng, 2011) or by introducing random noise in the input layer like in denoising autoencoders. Classical autoencoders aim to reconstruct the input data from that abstract representation, i.e. learn the identitiy function that transforms the input to the same output. However, if the goal is not reconstruction but generalization, autoencoders can become useful for many other problem domains. When stacked, the network can activate low level features and high level features, enabling robust classification. Or, as way to simplify training, extend a classical unsupervised autoencoder topology after training with a standard MLP. When training on a small labeled training dataset, a correlation between robust features and labels can be formed, thus creating a robust classifier (Hinton, Osindero, & Teh, 2006). In relation to ASR systems, autoencoders have been proven useful for speaker recognition (Heck, Konig, Kemal Sönmez, & Weintraub, 2000) and acoustic modelling (Feng, Zhang, & Glass, 2014).

Reinforcement Learning NN

To conclude the general four types of machine learning is reinforcement learning (RL) or *learning to act by trial and error*. RL requires a fitness/reward function such that it can maximize the total reward it receives over time. In essence devising the reward function can be seen as creating a generic labeler, because each unlabeled outcome result in an outcome that is scored (eg. good, neutral, bad). Various strategies have been created to implement RL in NN, such as ones that adapt its topology using the principles of evolution (Stanley & Miikkulainen, 2002) or its weights (Bing-Qiang, Guang-Yi, & Min, 2005) and (Coulom, 2002).

Training NN's using RL is tricky, in part due to finding a suitable reward function but also because in many situations a single input sample might not contain all the information to get the highest reward. For example, N-gram language models use the n-previous samples (a history) to reduce the possible search space and increase the likelihood for the remaining possiblities. Using RNN's this issue can be addressed, eg. for dialogue generation (Li, Monroe, Ritter, Galley, Gao, & Jurafsky, 2016). Still, RL in real-time dialogue systems can cause unexpected effects such as the Microsoft's Tay chatbot fail (Phrasee.co, 2016).

Current limitations of the state-of-the-art

With the state -of-the-art ASR systems having reached human parity and the continued effort of major research labs to improve Word-Error-Rate, achieving performance beyond human parity (< 4%) is on the near horizon. However for these ASR systems to perform more similar to humans the Switchboard and NIST dataset need to be replaced with more challenging datasets. These new dataset should relax constraints inherent to the previous datasets by addressing adverse conditions in which humans experience far less recognition issues such having a conversation with multiple speakers, regular and irregular backround noise, very near and remote audio speakers, heavily compressed speech signals, new words and names and finally generalize performance over to other languages than English.

The review of (Huang, Baker, & Reddy, 2014) mention six fundamental unsolved problems, not always specifically for the field of ASR:

- More data. Current systems are not nearly exposed to the same kind of sampling people routinely experience, ie. more speech, environments and modalities.
- Computing infrastructure. The use of GPU's is a significant advancement in recent years, but is not nearly enough compared to the capabilities of the brain.
- Portability and generalizability. Porting trained models for different languages or adapt a model (eg. to a speaker accent) with few data is still not possible, although the area of Machine Translation is very active.
- Unsupervised learning. The issue of knowing when something is learned and how the learned knowledge can be added to existing knowledge without any human intervention. This issue was also mentioned at one of the most important conferences for Machine Learning, NIPS 2016 (Cun, 2016).
- Having Socrates' wisdom. Being able to know when the system doesnt' know something. Related to this is when a result doesn't make sense while still being grammatically correct (eg. have a form of common knowledge).
- Dealing with uncertainties. The speech signal can be warped by background noise, room reverberation, multiple speakers, speaker quirks and technical factors like compression when using VOIP and many more.

2.2 Multimodal systems

2.2.1 Introduction

Humans have three common modalities or single independent sensory channels to perceive computer system output: visual, auditory and haptic (Wechsung, Engelbrecht, Kühnel, Möller, & Weiss, 2012) Other, more uncommon modalities are gustation, olfaction, and many more such as thermoception, magnetoception, hunger, thirst, time and pain. Multimodal systems combine one or more modalities. Six levels of cooperation between modalities can be defined (Grifioni, 2009).

Table 2-2 *Six levels of cooperation*

Equivalence	Information is presented in multiple ways and interpreted as same information
Redundancy	The same information is presented in multiple ways
Concurrency	Multiple modalities take in separate information that is not merged
Complimentary	Multiple modalities take in separate information that is merged
Specialisation	Specific information is always processed by a specific modality
Transfer	Information produced by modality A is consumed by modality B

In ASR research a number of multimodal systems have been published about that exploit multimodal information to combat the issues mentioned earlier such as noise, speaker identification and large vocabularies. In this thesis we explore two kinds of unimodal sources (audio and video) and three kinds of input (speech, objects and gestures) so the following selection is limited to multimodal Audio-Visual (AV) systems possessing two or more of these inputs.

Audio Video ASR systems

Combining speech with video of the movement of the speakers's mouth has been proven to significantly improve ASR performance (Potamianos, Neti, Gravier, Garg, & Senior, 2003). For example (Bregler & Konig, 1994) already demonstrate a WER score of 46% using its Delta-Lips system versus 67.3% using the audio only system. The work of (Tamura, Iwano, & Furui, 2005) hint that this improvement does not apply to English solemnly, but can also be found in Japanese, again for a similar goal as ours: practical robust speech recognition in the presence of additive noise ("the cocktail party effect"). Fusing more modalities such as 3D facial depth information using a Microsoft Kinect sensor result in further improvement over audio only or audio+video ASR recognition (Galatas, Potamianos, & Makedon, 2012).

Various solutions have been addresss for the asynchrounously problem (see (Estellers & Thiran, 2010) and (Gravier, Potamianos, & Neti, 2007) as visual speech activity precedes the audio signal by as much as 120 ms example (Bregler & Konig, 1994). The current state-of-the-art in lipreading ASR system, LipNet (Assael, Shillingford, Whiteson, & Freitas, 2017) uses spatiotemporal CNN's and bidirectional GRU (type of RNN) to overcome this problem. LipNet achieves 95.2% accuracy on the GRID corpus, outperforming hearing-impaired lipreaders (52.3%) and the previous record (86.4%) by (Gergen, Zeiler, Abdelaziz, Nickel, & Kolossa, 2016).

Exploiting the video modality to aid a speech recognizer is not exclusive to the domain of lip reading ASR systems. For example, by recognizing action events in video, (Fleischman & Roy, 2008)ASRs can be primed in its recognition (increase likelihood for certain words) and demonstrate improved automatic video transcription of particular sports events over audio only. (Roy & Mukherjee, 2003) demonstrate that simple and more complex natural language

phrases such as "*the red block*" or "*the left most large one*" can be jointly learned with the object of interest's visual features such as shape and color. They also show an eight person average WER improvement of 7.6% over audio only for simple sentences. For complex sentences this was 10,1%. Both were achieved with an early fusion strategy of visual context in the speech recognition process.

Intent (action) classifying ASR system

Using computer vision many features can be extracted from the video modality, especially when using 3D sensors. Skeleton tracking (Shotton, et al., 2011) and hand tracking (Sharp, et al., 2015) invariant of lighting conditions enable accurate tracking of human interaction with objects. (Chol Song & Kautz, 2012) show that extracted language and gesture features can be fused to create a model for activity recognition. They trained the model by demonstrating activities such as making tea, but did not use a speech recognizer to convert text-to-speech.

(Rossiter, 2011) did use a speech recognizer, as well as a gesture recognizer and fused both using a MLP type neural network. In their "Wizard of Oz" experiment they captured speech and 3D gestures from particpants that were tasked to guide an AIBO robot along a path as natural as possible. Their goal was similar to ours - train a fused model that has better recognition capabilites than the individual modalities. As speech and gestures are not as tightly coupled as other modalities (such as speech and lip movement) they used a late (decision based) fusion approach.

With the launch of the Microsoft Kinect 1.0 released in november 2010 an affordable 3D sensor with microphone array was released. This led to new 3D gesture research (Nguyen-Duc-Thanh, Stonier, Lee, & Kim, 2011) speech-gesture (eg. (Song, Kautz, Lee, & Luo, 2012) as well as this thesis with feature speech, gesture and object modalities.

In literature other modalities have proven to be succesful in improving speech recognizers, notably geolocation based ASRs. For instance, use location information to prime the language model for nearby places (Chelba, Zhang, & Hall, 2015)or use location information to prime the acoustic model for certain accents (Ye, Liu, & Gong, 2016).

2.2.2 Multimodal fusion issues and strategies

Fusing modalities is not trivial, even if the level of cooperation between modalities is clear. (Atrey, Hossain, El Saddik, & Kankanhalli, 2010) in their comprehensive overview on multimodal fusion state the following practical and theoretical issues with multimodal fusion:

- Multiple media formats and rates (eg. frame rate). The fusion strategy needs to address this with some asynchronous solution.
- Processing time of media streams migth be dissimilar (eg. 4K video over mono audio). The fusion strategy needs to address this with some asynchronous solution.
- Modalities might be corrrelated or independent (eg. speech and lip movement). The fusion strategy needs to be designed such that it exploits these correlations.
- Modality fusion might be context or task dependent (eg. rely on audio in darkness or extract cry emotion from audio over video). The fusion strategy needs to analyse and weight modalities dynamically. When relying on the wrong modality, fusion can go "catastrophicaly wrong" (Movellan & Mineiro, 1998).
- Capture and processing media streams might involve costs (eg. best modality migth be too costly or not always available). The fusion strategy needs to have a contingency plan for unavailability or alternative sensor fusion.

Multimodal fusion strategies

The multimodal systems discussed in 2.1 use various fusion strategies to solve the issues mentioned above. In general these are all related to the question "*when, how and what to fuse*". Key parameters in these strategies all are:

- Fusion architecture (ie. *level of abstraction* and *cooperation*)
- Multimodal learning scenario (ie. *when and what to fuse*)
- Fusion method (ie. *how to fuse*)

Fusion architectures

One of the earliest considerations when fusing multiple modalities is considering the fusion architecture (Pradeep, 2010). Three variants exist, with additional mixture of the three:

- **Early fusion (feature fusion, fig 2-7).** This architecture is the most widely used and fuses modalities at the input level as features, hence the name early or feature fusion. It is the most naive approach, although with careful feature engineering strong multimodal systems can be devised (as mentioned in section 2.1). Also, although it is easier to learn with a single combined feature vector early on, it is harder to do the synchronisation when modalities are out-of-sync.
- **Late fusion (decision fusion, fig. 2-8).** This architecture fuses modalities in the semantic space by first processing the features into a score or decision, often with a confidence value. This type of architecture has disadvantage that it might miss the correlations between modalities as they are abstracted by the decision analyser. On the other hand the representation of the decision analysers are often equal (in contrast to the many formats at the feature level), simplifying fusion and improving inference transparancy.
- **Interactive fusion (fig 2-9) .** In this architecture fusion takes place on intermediate results, dynamically. This type of archicture's main advantage is enabling context or task dependent fusion by first analysing the features or decision. Analysing content and generating a model for dynamic classifcation is not trivial, hence the limited availability of interactive fusion systems.
- **Hybrid fusion.** This type of fusion combines all three architectures in one model. Theoretically it could benefit from all early and late advantages at the price of added complexity.

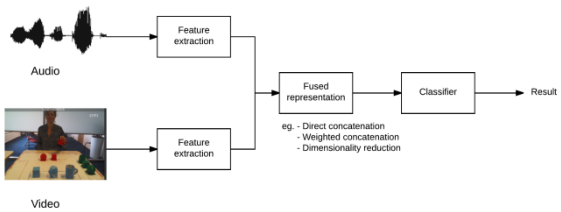


Fig 2-7 *Early fusion*

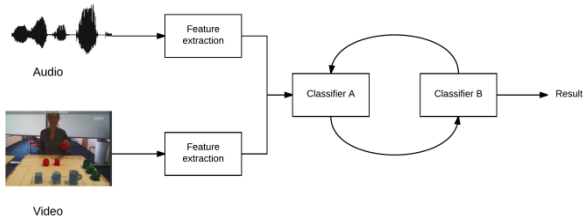


Fig 2-9 *Interactive fusion*

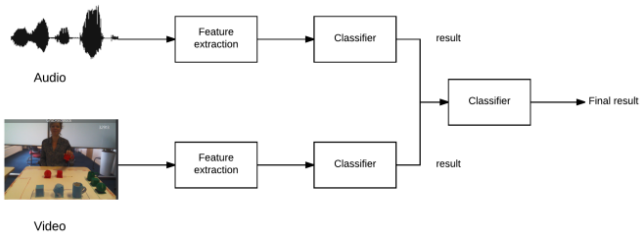


Fig 2-8 *Late fusion*

Multimodal learning scenario

Multimodal learning is the process of correlating information from multiple sources (Ngiam, Khosla, Kim, Nam, Lee, & Ng, 2011). This process involves three typical machine learning phases: feature learning, model training and model evaluation. Traditionally in multimodal fusion, all sources are available at all three phases. For completion we enlist some other variants described in literature in Fig 2-10. Next to multimodal fusion, shared representation learning is very interesting for building better ASR systems as the video signal is only present during training and can be omitted during evalation (like in a telephone call).

In this thesis we use grounded learning which is a form of multimodal fusion. We will learn a joint model of language and perception somewhat similar to (Roy & Mukherjee, 2003) and (Matuszek, Fitzgerald, Zettlemoyer, Bo, & Fox, 2012) as decribed in detail in chapter 3.

	Feature Learning	Supervised Training	Testing
Classic Deep Learning	Audio	Audio	Audio
	Video	Video	Video
Multimodal Fusion	A + V	A + V	A + V
Cross Modality Learning	A + V	Video	Video
	A + V	Audio	Audio
Shared Representation Learning	A + V	Audio	Video
	A + V	Video	Audio

Fig 2-10 *Multimodal learning scenario's*

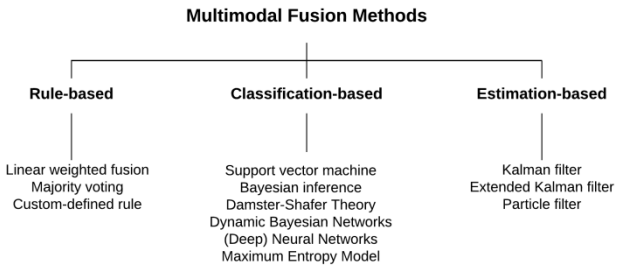


Fig 2-11 *Fusion methods*

Fusion Methods

Many fusion methods have been proposed, see (Atrey, Hossain, El Saddik, & Kankanhalli, 2010) for an extensive overview. The fusion methods can be divided in three classes (see Fig. 2-11):

- Rule-based
- Classification-based
- Estimation-based

This thesis will only explore the (deep) neural network, a classification-based fusion method.

2.3 Automatic Speech Understanding

2.3.1 Introduction

Natural Language Understanding (NLU), a subfield of Natural Language Processing (NLP), is a relative new field for computer scientists with only about 50 years of active research, dating back to the early ASR research. The field can be approached from many different angles such as linguistics, (evolutionairy) biology, psychology, neurology, philosophy, pedagogy and computer science - making it a true interdisciplinairy field. Within computer science literature NLU is labelled as *Automatic Speech Understanding (ASU)*, a term we prefer as it expands the goal of ASR and distinguishes from *Computational Semantics* (CS), a term from the linguistics field. Prior to this thesis we published an elaborate survey on the components involved in artificial NLU systems (Bos, 2012). This section picks few parts from that paper, although some updated to 2016 insights.

Our working definition for ASU:

"*ASU is the process of automatically converting spoken language (speech) to meaning structures (eg. grounded words, reponses that make sense in relation to queries)*"

Compared to ASR system, ASU systems focusses on making sense beyond the syntactic level, both in isolation (one-shot Q&A) and in dialogue (continuous Q&A). This semantic dimension can be approached by knowledge representation (MacCartney & Potts, 2016)(Fig 2-12) or by knowledge level (Fig 2-13) (Allen, 1987). Traditionally ASR focuses on the top four while ASU focuses on the bottom four. The divide-and-conquer approach of 8 levels is still relevant today (eg. NLU in Healthcare) (Iroju & Olaleke, 2015) (Cambria & White, 2014) prefers to stop at the pragmatics level by including discourse and common sense analysis within that level and roughly divide NLU in three develoment stages (Fig 2-14).

sentiment analysis	continuous	scalars	
vector space models		vectors / topic distributions	
relation extraction	discrete	relation instances / database triples	(Larry Page, founder, Google) (Google, located in, Mountain View)
semantic parsing		logical forms / other rich structures	argmax(λx.state(x), λx.size(x))

Fig 2-12 *NLU organisation in representations*

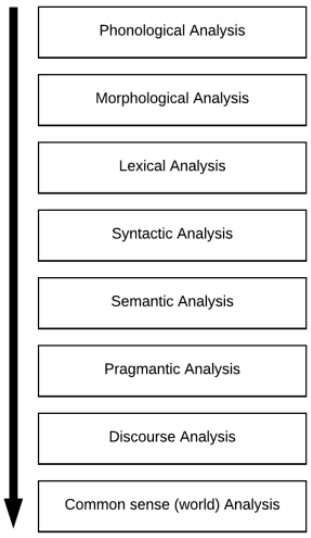


Fig 2-13 *NLU organisation in levels*

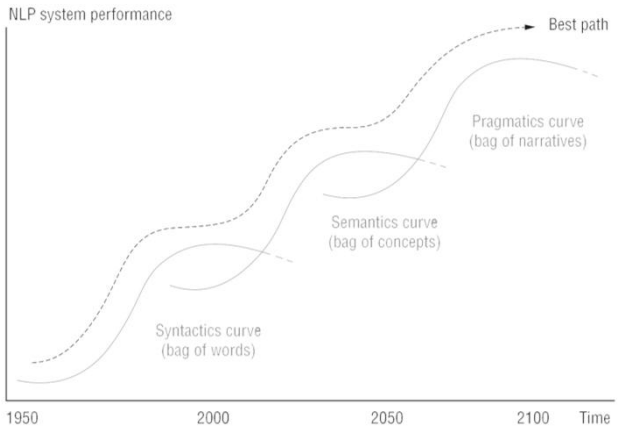


Fig 2-14 *NLU development stages and timeline*

Characteristics for ASU systems

There are many issues that make the process of converting speech-to-semantic-text hard, all related to how language encapsulates meaning using symbols (words) and how the brain processes it to an interpretation. Each of the NLU levels mentioned above have its challenges to solve. Some of these issues are (for more see (Bos, 2012).

Table 2-3 *ASU Characteristics*

Modelling Multimodal context	Problem about what part of the world needs to be included in a knowledge representation and how to structure it
Learning Semantic units	Problem of learning that the relation between Form and Meaning, eg. two words, can express one or more new meanings when combined.
Generalizability	Problem of limited or no links to previously learned knowledge when switching to new domains.
Modelling Embodiment	Problem of modelling human sensing, processing and actuating. Time constraints for parsing, sensor sensitivity of the ears, nose, etc and parallel processing of form and meaning concurrently in the brain and acutator states of the hand and legs yield unique parses of speech-to-semantic text.

Modelling ASU systems

ASU systems usually processes natural language input to some knowledge structure output, ie. speech-to-semantic-text. Fig 2-15 displays how this process is usally modelled. A semantic theory is not visable in the model but is affecting both the interaction flows (making the flow not neccesarily serial and hierarchical) as well as the actual processing (parser, inference, etc).

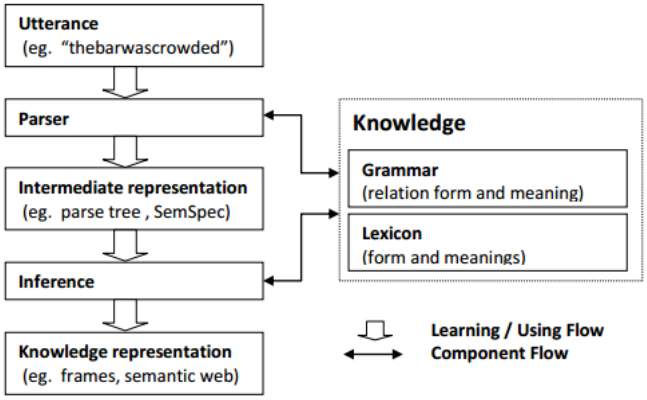


Fig. 2-15 *General NLU architecture*

This conversion process often includes the following components:

- Lexicon. The dictionary, a collection of words/symbols that can exist in a language.
- Grammar. Rules (patterns) to break the sentence into an internal representation. Parser. Procedure (often called search strategy) to segment the sentence, such that all possible structures that can be derived from the input sentence are found.
- Logical Inference. While the parser is basically a dumb pattern recognizer or pattern search and matching mechanism, the inference engine is the actual "intelligent" mechanism. Its design is based on the semantic theory. A good engine would have two functions:
    - Disambiguation (pattern selection). Make a choice (disambigue) between intermediate structures (if there are multiple structures)
    - Pattern finding and learning. The learning creates new grammar rules for the grammar and symbols for the lexicon
- Knowledge representation. The intermediate or internal representation (analogous to human thoughts) such as semantic web, frames, graphs and many others.
- Semantic Theory. Theory on how the knowledge such as the grammar and lexicon are acquired, structured and used

2.3.2 A brief review: from Machine Translation to Intelligent Assistant

The earliest NLU research was done in the early 1950's and focussed on Machine Translation (MT) from Russian to English. The journal Mechanical Translation, ancestor of Computational Linguistics began publication in 1954 (Sparck Jones, 2001). Possibly one of the earliest succesful NLU systems was Weizenbaum's Question&Answer (Q&A) system "Eliza" (Weizenbaum, 1966). The technology behind it was simple, basic scripting (template matching) and some rules for the rewriting of questions like a Rogerian psychotherapist (open ended questions) – "I don't feel well" becomes "Can you elaborate on not feeling well?". The system was a shocking success, some people chatting with it for more then half an hour without noticing it was a computer program. Another famous program was SHRDLU (Winograd, 1975) possibly the first embodied NLU system. It was able to have a conversation in a restricted domain, the "block world" and was able to move blocks and reason about them. It was able to process instructions like "put the green block on the red block" and questions like "did you pick up anything before the green block?". Since then (over 50 years ago, see (Cambria & White, 2014) for a thorough overview) much progress has been made in the fields of phonetics, morphology and syntactics, the domain of ASR systems. The rise of the WWW and availability of large amounts of text data, combined with a demand for insight caused much innovation in the syntactics domain. Word co-occurance models (often performed on keywords such as nouns or verbs) and semantic similarity and topical similarity operate on the lexical level (Ferreira Junior, 2013).

In comparison, until 2010 little progress has been made in the fields of (compositional) semantics, pragmatics, discourse and world knowledge. The development of ASU systems roughly follow the same technology timeline as with ASR systems. First, rule-based (template matching) and first-order logic systems, then probabilistic systems and currently neural network based systems. In the last six years, with the advent of deep neural networks and focus on big data and machine learning in general, simple Q&A systems evolved to sophisticated architectures. In 2011 IBM's Watson (fig. 2-16) caused the first NLU breakthrough in decades, when it bested the top two human players in a game of *Jeopardy!*, which requires deep understanding of the question as well as respond with a high confidence aswer in under 3 seconds (Ferrucci, et al., 2010). Unfortunately no speech recognizer was used at the time as the question was fed as plain text. The huge amount (100+) of machine learning algorithms used were rule-based and probabilistic. Feature engineering was key part of its success, as one of its researchers mentions: "...*the vast majority of the work that was done on the Watson project focuses on techniques for finding candidate answers and computing feature values for those candidate answers*".[5]

[5] https://www.quora.com/Is-IBMs-Watson-an-Expert-System

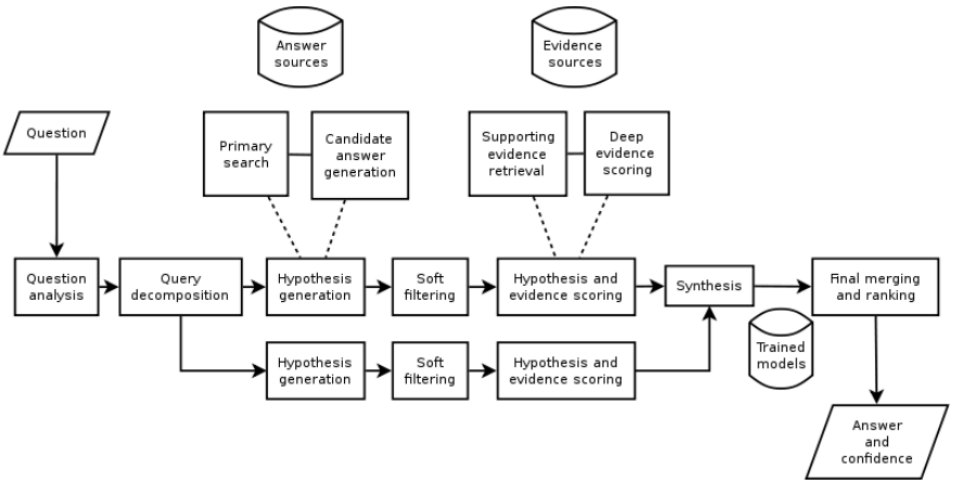


Fig. 2-16 *Watson's DeepQA Architecture*

2011 also marked the rise of intelligent assistants such as Apple's Siri, Microsoft's Cortana, Google Now and Amazon's Alexa. These NLU systems track both question and context such as location, time, user preference and usage history. Furthermore they respond with helpful actions such as setting alarms and look up information. The conversations possible are still limited (domain specific) and shallow (follow up questions are hard), as the dialogue and semantic dimension in these evolved Q&A systems is limited and often unimodal (fully text based). With curated resources such as WolframAlpha and Wikipedia these systems can answer questions in more domains but lengthy and insightful conversations are currently not possible.

The current most advanced conversationional bot or "chatbot" is “Mitsuku” from Steve Worswick winning the 2016 Loebner Prize. The Loebner Prize is an annual contest where several judges need to be convinced their chat partner is human and not a computer program – the famous *Turing Test* (Turing, 1950). It should be noted that winning the Turing Test – which is essentially a mimicing game – is only the starting point of good NLU systems and intelligent systems, since it is a one-sided test. The techniques fully exploit the expectations humans have when chatting. When the chatbot is pretending to be a child with English as its second language, odd responses are more common. Also, facial and body language and contextual audio (eg. walking outside, breathing heavily) is not transmitted which are important cues in real life. The inventor of the term AI, John McCarthy mentions on his website[6], that passing the test “*does not really test whether a machine or computer program is intelligent, only that it is able to fool a human*” (paraphrase) – something that Eliza did without much effort, if the human is caught off guard. McCarthy also mentions that AI reaching human level intelligence requires new paradigms and new approaches to escape the current “in-the-box-thinking”. The current widely popular “more data” approach is the easy approach, however human intelligence is not about quantity – humans are not omnipotent - but about qualitative usage of data.

[6] http://www-formal.stanford.edu/jmc/whatisai/node1.html, last visited Januari 2017

3

Methodology

With an introduction to all the relevant components for our ASU system and multimodal fusion in general we discuss how we set up our experiment to collect and analyse data using this system.

3.1 Introduction

"Research is about making knowledge"[7]. For this thesis we choose a quantitative research methodology to achieve this knowledge. Specifically, we follow the classical experimental design process shown in Fig. 3-1 and use a deductive reasoning approach starting from our working hypothesis:

"Fusing classified objects and gestures from a 3D camera with speech have a beneficial effect on the WER performance of an Automatic Speech Recognizer under increasing Gaussian white noise condition from 20 dB up to -5 dB SnR compared to speech only.

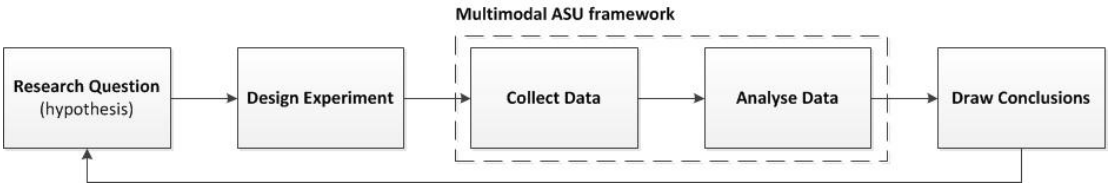


Fig 3-1 *Experimental design process supported by proposed multimodal ASU framework*

To test this hypothesis we designed an experiment and methodology to collect and analyse data, described in chapter 3.2 and 3.3. The framework to support and process the data is described in chapter 4.2.

3.2 Experiment

To train the components in our multimodal architecture such as the neural network, we need to collect a large amount of data. We devised an experiment and developed a framework for synchronous multimodal data recording and labeling data. In total 25 particpants were recorded, aged between aged 23-53 years, predominantly non-English native speakers and about 1:3 female to male ratio. All recordings took place at the Geodan Amsterdam Research Lab, The Netherlands.

[7] Quote from Ivan Hofsajer from his EdX MOOC "*Research Methods: An Engineering Approach*"

We chose this grounded language learning experiment as it simulates a human teacher-learner setting. The particpant demonstrates the opposite (learning) party word usage in physical context as realistic as possible, so with adult-directed speech and unconstrained gestures (eg. some participants use a pull gesture for grabbing occasionaly). We strategically placed the camera closeby and in front of the particpant such that we can capture relatively clean data with few occlusion or noise due to distance while participant focusses on teaching words and gestures to the screen in front of the particpant on eye level.

Grounded language learning experiments with blocks have been done before, eg. (Winograd, 1975) as have experiments with learning by example (Roy, 2002).

3.2.1 Setup

The experiment took place in an office workspace. The equipment in section 3.2.2 were placed on the table as shown in Fig. 3-2 and 3-3. The setup is designed to have a direct line of sight with the hands to approximately detect the grabbing and releasing of an object.

Figure 3-2 *Render of the experimental setup*

Figure 3-3 *Actual setup at the Geodan Research Lab in Amsterdam*

Engineering the recording software of the experiment proved to be tricky as the raw Kinect sensor streams saturate the 60 MB/s (480 Mbit/s) USB 2.0 bus:

(30 Hz) * (640 * 480 pixels) * (4 bytes per color pixel) = **36,9 MB/s +**
(30 Hz) * (640 * 480 pixels) * (2 bytes per depth pixel) = **18,4 MB/s +**
(16000 Hz) * (4 channels) * (3 bytes per audio sample) = **0,2 MB/s**

= 55.5 MB/s, leaving just 4.5 MB/s for overhead

With such small margin for error, the recording software is required to capture audio and video frames, copy it to memory and release the lock on the sensor as soon as possible, in order to prevent frame drops. With no compression, the recording software then needs to sustainly write memory to a physical disk (magnetic or SSD) with a capability of writing speeds of at least 60 MB/s in order to prevent buffer overflows.

3.2.2 Task, Variables, Instruments and Measurements

Task

Each participant is instructed to execute a single task:

"teach the virtual person (in adult directed speech) across the table the action on the screen by performing the gesture while simultaneously saying it. "

An instruction is phrased like "*say and do the following*" followed up with another screen stating "*I grab the red block*". All instructions are short (a single, one handed gesture) and involve only one object. One session involves the performance of 54 actions, with an additional 9 miscellaneous gesture-object-speech actions used for future research.

Objects

We modeled three 3D objects (block, ball, cup) using the open source modelling tool Blender. The models were printed in three colors (red, green, blue) on a Felix 3.0 3D printer using PLA filament. We used 40% infill to give the model some weight. Each model has a square base of 6.5cm and average height of 7.5cm. For the first 10 experiments we used three objects of similar shape and dimension, coated in blue paper as our blue filament was unavailable.

Variables

Our working hypothesis has two independent variables that are manipulated

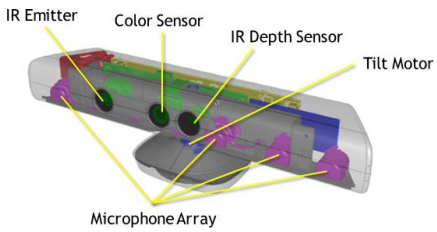
- *Modality*: audio only, video only, multimodal (fused audio+video)
- *Noise level*: 20, 10, 5, -5 dB SnR (for audio)

and one dependent variable that is observed:

- *WER performance*: in scale 0-100% error

Instruments

The primary instrument to record both 3D video data and audio was the Microsoft Kinect version 1 (Fig. 3-4) . This sensor is capable of providing both color + depth videoframes and multi channel audio data. Next to raw audio and video streams its Software Development Kit (SDK) offers low performance cost feature streams like *player identification* and *3D skeleton tracking*. The skeleton tracking stream is able to a track human joints in 3D space and is used as the input stream for our gesture recognizer. The raw output of sensor can be seen in Fig. 3-5.



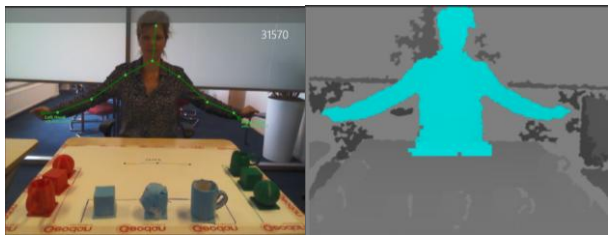


Figure 3-4 *Sensors in the Kinect V1* **Figure 3-5** *Raw data (RGB+skeleton left, Depth right)*

The computer used to process the Kinect sensor and the recording framework was a custom build quad core i7 4770K CPU with 32 GB of RAM, 1 TB SSD and a GTX Titan GPU. The machine ran Windows 8.1 Professional using the .Net 4.5 framework and Visual Studio 2015 Enterprise.

Measurements

From each particpant the following data was recorded:

- Voice data in 32 bit PCM encoded format and sampled at 16Hz
- Skeletal data of the joints at 30Hz
- Color frames in VGA (640x480) resolution in RGBA (red, green, blue, alpha) format at 30hz
- Depth frames in VGA resolution in 16 bit (13 for depth, 3 for player identification) at 30hz.

3.2.3 Procedure

The general flow of a single experiment session is shown in Fig. 3.2-6. We tested this workflow and optimized our software settings using a small pilot study with 3 participants not in our list of 25 participants.

Figure 3-6 *Experiment stages*

Welcome (1 minute)

Participants were welcomed and seated in the particpant chair in a private room. Next, a small briefing about the experiment was given. In this briefing we disclosed an estimated duration, the purpose of the drawings on the table and kindly asked to switch off their smartphone and follow the video instructies carefully. The administrator then closed the door (making the room silent to external noise) and started the video instructions.

Video instructions (1 minute)

The video instructions demonstrate a recorded example of a gesture and speech action in sync performed by a participant. Right after the instructions, the main experiment started. The participant was requested to carefully execute the actions shown on the monitor.

Main experiment (5 minutes)

The main experiment was divided in a fixed three “color” stage order (red -> green -> blue). Each color stage contained 9 actions (see Table 3.1). First the participant was asked to grab three objects from the same color and place them on the marked area in front of the participant in random order. Then 9 actions were requested one at a time. Unknown to the subject, they determined the pace and as soon as the action was performed the next instruction was given with no pauses. Only after 9 actions a small 10 second break was given. The same 9 actions were requested but in a different order, followed by a request to return the three objects to the colored region outside the workspace. This was repeated for the other 2 colors, although in the first 10 experiments the blue object interactions was limited to 9 instead of 18.

Table 3.1 *Overview of all 27 actions, with 9 actions for each color stage*

	Red	Green	Blue
Block	*Grab/Push/Pull*	*Grab/Push/Pull*	*Grab/Push/Pull*
Ball	*Grab/Push/Pull*	*Grab/Push/Pull*	*Grab/Push/Pull*
Cup	*Grab/Push/Pull*	*Grab/Push/Pull*	*Grab/Push/Pull*

eg. "*I **grab** the **red block***"

Side experiment (1 minute)

During the test run of the experiment it became apparent that 54 actions in 5 minutes resulted in bad task execution, most likely due to the repetitive character and simplicity of the gesture-object-speech actions. Ten second breaks were introduced between 9 interactions as well as two side experiments to take the particpants mind of the main task as well as record data for future experiments. The first side experiment, between the first and second color stage, requested to say and do contradictory things, in increasing complexity. The second side experiment was performed after the last color stage and requested the player to do novel and unexpected gestures like pointing at a block not in the workspace or stacking objects.

Questionnaire (2 minutes)

After the experiment a short questionnaire was held, consisting of 12 questions: 6 demographic questions about individual characteristics (name, age, gender, educational background, color blindness, native tongue(s)) and 6 subjective questions about their estimated command over English pronunciation and questions about the performed tasks.

3.3 Data analysis

For this quantitative study all recorded 25 sessions were postprocessed using custom tools, analysed using the WER metric and compared to the audio-only baseline. This subsection describes how this was done.

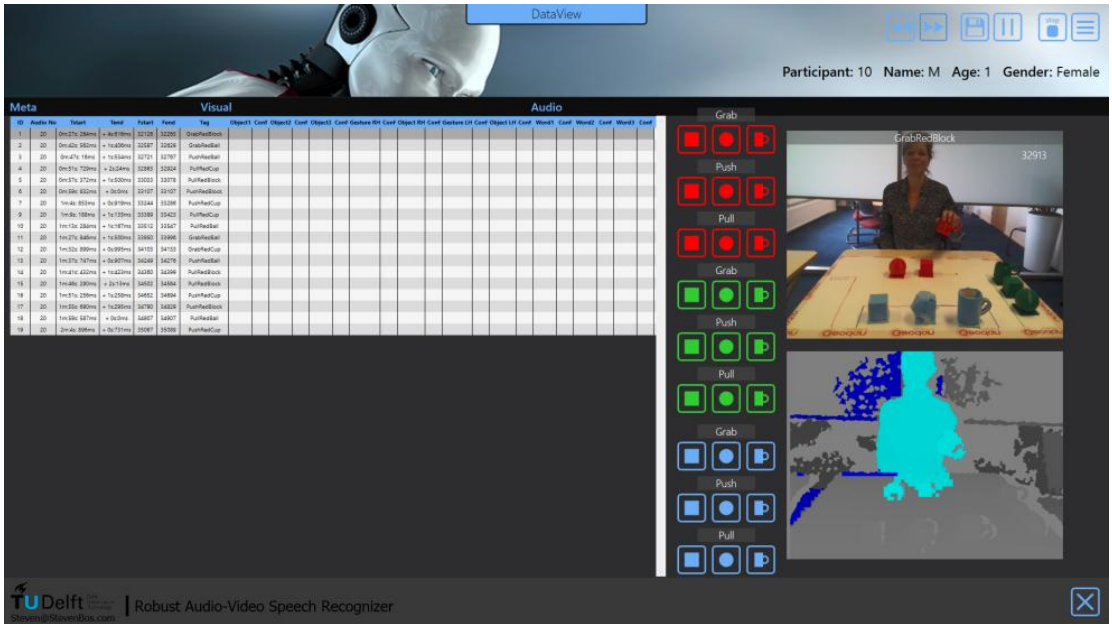


Figure 3-7 *Tool for labelling groundtruth*

Postprocessing: Labelling the groundtruth and generating Gaussian white noise

The recorded dataset was labelled manually to obtain the ground truth. All of the 54 samples of every session was labelled with information on Tstart, Tend and Tag using tools (see Fig. 3-7) designed specifically for this experiment. Wrong samples, those with a mismatch in gesture-object and speech (eg. *grab a red cup* when saying *"I grab a red block")* were not labelled and discarded. However samples with a mismatch in asked action vs executed action (with gesture-object-speech in sync) were labelled accordingly as the experiment was not about measuring task execution, but about collecting data in a way that gave a more or less uniform distribution across all 27 classes for training our models.

To measure ASR performance on 4 SnR levels, raw sensor data was postprocessed with a custom Gaussian white noise generator. Each audio frame of ~130ms got analysed in real-time with noise added to the frame before sending it to the speech recognizer using the formula:

[1] $$SNR_{dB} = 10 \log_{10}\left(\frac{P_{signal}}{P_{noise}}\right)$$

With P as *average power* and both signal and noise representing amplitudes in 16 bit (signed short) bandwidth. When *average power* is written as root mean square amplitude or simply the mean of that signal the formula becomes

Metrics

To evaluate ASR performance, the industry standard Word Error Rate (WER) and Word Accuracy (WA) is used as wel as Precision/Recall (F1-score) and Confusion Matrix.

WER

WER computes the minimum edit distance or *effort* that need to be done to convert the input (recognized) sentence to the (correct) reference (groundtruth) sentence. With effort is meant the basic transformations *substitutions*, *deletions* and *insertions*. The WER is thus calculated for each sample as

[2] $$WER = \frac{I+S+D}{N}$$

Where S = *#substitutions*, D = *#deletions*, I = *#insertions*, N = *#words in reference sentence*

The average WER for all three modality levels (audio only, video only, multimodal) is computed over the validation set of $\frac{1}{5}$ of 25 particpants, such that no training samples of that participant are provided to test the recognizers. Furthermore, the average WER for each modality is computed for each of the five (20,10,5,0,-5) noise levels.

[3] $$\overline{WER}_{modality,snr} = \sqrt{\left(\sum_{p=1}^{8}\sum_{s}^{S}\frac{I+S+D}{N}\right)^2}$$

Where p ∈ $\{participants\}$ and s ∈ $\{valid\ recorded\ samples\ of\ participant\ p\}$

The WER scores are finally converted to a more intuitive representation (low percentage equals low performance) using the industry standard (Jurafsky & Martin, 2000)Word Accuracy (WA) metric for ASR performance using the formula

[4] $$Wacc = 1 - WER$$

Precicion/Recall

To measure system performance we use an evaluation measure often used in information retrieval. By calculting precision and recall we can measure the exactness (quality) and completeness (quantity) of the system. They are defines as:

[5] $$Precision = \frac{\#retrieved\ and\ relevant\ documents}{\#all\ relevant\ documents}$$

[6] $$Recall = \frac{\#retrieved\ and\ relevant\ documents}{\#retrieved\ \ documents}$$

Formula [6] can be rewritten in terms of Word Accuracy when I = 0 (no insertions, as is the case due to our DNN returning one of 27 fixed sentences) as

[7] $$Wacc = \frac{N-S-D-I}{N}$$

[8] $$Wacc = \frac{N-S-D}{N}$$

Confusion Matrices

Finally we use confusion matrices to plot and analyse the performance of our semantic analyser. These form the heart of our analysis in optimizing our recognizers. We define our 27x27 classification confusion matrix:

- 27 actual classes
- 27 predicted classes

WER scoring process

Scoring the ASU system can be done in various ways. Since we use asynchronous data we use sliding windows to buffer events before sending it to the detection stream to compare to the groundtruth, as can be seen in figure 3-8. For each groundtruth we also use a small sliding window buffer as we are working with real data and events can occur just before and after. Note that if we score neural network performance, each detected interaction event is instantly processed by the deep neural network such that that timing is identical to without. We guarantee this behaviour since we process all interaction events in an offline phase (real-time processing of the neural network takes 1+ seconds). Using this scoring method makes sure we never miss our groundtruth. Since the goal was not to optimize for false positives, we ignore overactive recognizers.

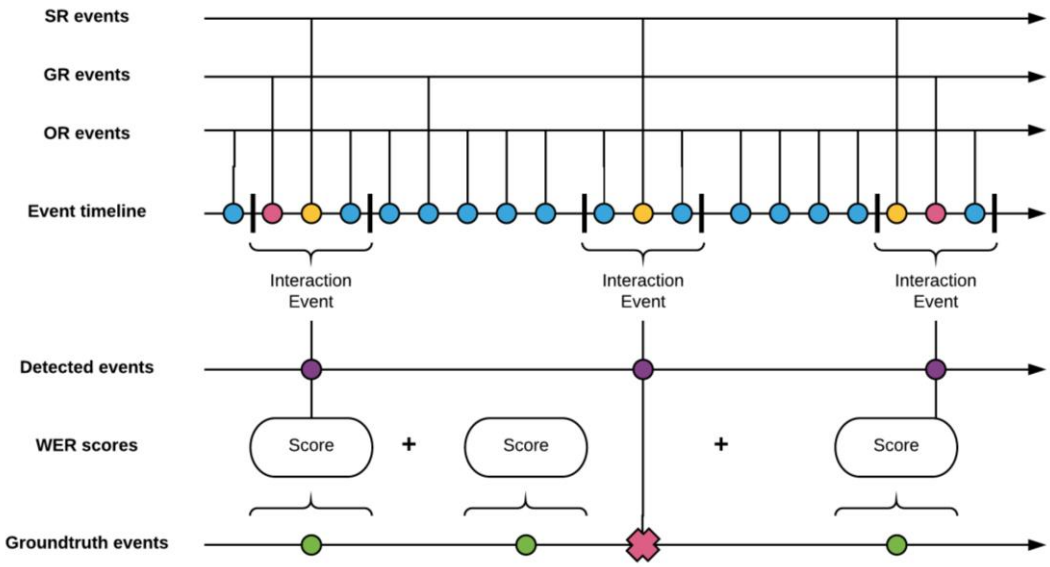


Figure 3-8 *WER scoring process*

Benchmarks

Although much benchmarks have been published on ASR (audio only), CV (gesture and/or object tracking) and multimodal ASR systems using the lip visual feature, no benchmark exist for ASR performance fused with gesture and object tracking, which is the prime reasons we collected our own data. As mentionded earlier in this chapter, we didn't choose nor developed state-of-the-art recognizers, hence comparing them to the state-of-the-art would make little sense as no results have been published with our data. The measured *audio only* and *video only* baselines provide reference to our specific multimodal ASR system as we train it with our data.

Multimodal ASR benchmarks by (Galatas, Potamianos, & Makedon, 2012) and (Gravier, Axelrod, Potamianos, & Neti, 2002) and (Papandreou, Katsamanis, Pitsikalis, & Maragos, 2008) (fig 3-9) show that significant performance improvements can be achieved when fusing modalities, which is what this thesis pursuits. We accept our working hypothesis when we can prove that the multimodal approach is around 5 dB or better compared to audio only.

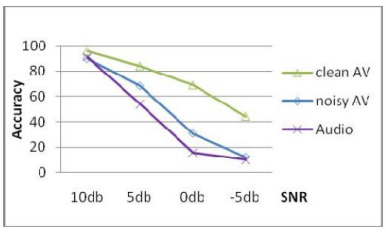


Fig 3-9. *Improvements due to MM fusion*

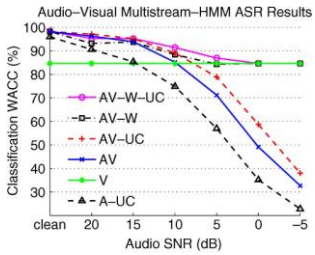


Fig 3-10. *Ideal system performance (purple line)*

This thesis aims to replicate a performance plot such as shown in Fig. 3-10 (Papandreou, Katsamanis, Pitsikalis, & Maragos, 2008) where multimodal sources are always equal or better than unimodal sources under noisy conditions.

4

Results

4.1 Dataset

The recorded and labelled audio-video dataset has the following characteristics:

General
- Size: 600 GB
- Duration 2,5 Hours

Modalities
- 3D video (2.5D RGB pointcloud)
- Multichannel audio (single speaker speech)
- 3D Skeleton (joints of upper body)

Groundtruth
- Speakers: 25
- Total categories: 27
- Average samples per speaker: 44
- Total samples: 1206
- Sample distribution over categories:

	Red block	Red cup	Red ball	Green block	Green cup	Green ball	Blue block	Blue cup	Blue ball
Grab	43	49	47	47	49	48	39	41	40
Push	46	50	48	47	51	50	37	39	42
Pull	42	49	50	47	48	48	36	36	37

The questionairres resulted in following speaker characteristics:

Objective
- Average age: 33,5 (range 23-53)
- Males: 17
- Females: 8
- English as native (first) language: 0
- Color blind: 0

Subjective
- Command over English (expert, advanced, beginner): 24%, 64%, 12%
- Quality of instructions (clear, somewhat, not clear): 92%, 8%, 0%
- Difficulty of experiment (too easy, somewhat, too difficult): 8% / 92% / 0%
- Duration of experiment (fine, somewhat too long, too long): 88%, 12%, 0%

4.2 Multimodal ASU Framework

To accommodate multimodal data and deal with the multimodal fusion issues we propose the *multimodal automatic speech understanding* (MASU) framework (Fig. 4-1). The scope of this framework extends beyond ASR by trying to understand speech using the correlation between audio (eg. words) and other contexts such as video (eg. objects, gestures, events), hence the name ASU.

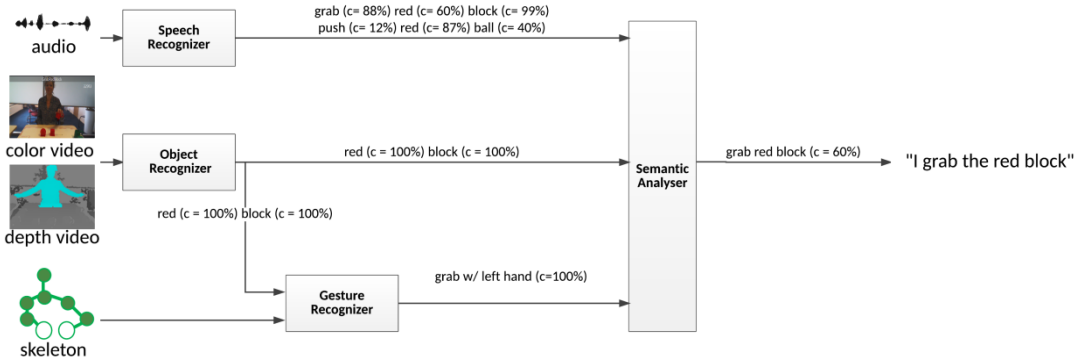


Fig 4-1 *Proposed multimodal ASU Framework*

Multimodal fusion

All multimodal fusion issues mentioned in chapter 2.2 are addressed in the MASU framework:

- *Early fusion*, fusion at the feature level, is used when there is clear correlation between data streams suchs as skeleton data (gestures) are correlated with 3D video data (objects)
- *Late fusion*, fusion at decision level, is done in all other cases often with confidence scores
- *Synchronisation* is done using a temporal buffer (a sliding window) and is inspired by how children learn and use words. According to (Tomasello, 2008) children have a small window where events can take place before and after the associated utterance.
- *Modality fusion* is done by processing the fused vector of decisions in the Semantic Analyser component using a deep neural network

The framework is classified as a hybrid fusion framework due to its mixture of early and late fusion. A thresholding strategy can be used in each component to reduce false positives.

Implementation

We constructed a workflow and an implementation (fig 4-2) using the framework to record/replay data, analyse and label data as well as train and test models. The framework is modular with API's for each component. It is designed to prototype and playtest simple Automatic Speech Understanding applications by swapping the recognizers for a new version or different type. After training with multimodal data, a joint audio-visual model of language is build such that new applications that dont have video input can still benefit from it. The multi-threaded framework is written in the C# .NET programming language using a WPF

(XAML) user interface. The only model that was not trained in the framework (but is tested in) were the deep learning models using the CNTK machine learning framework (Agarwal, et al., 2014) .At the time of writing only c++ and Python binding were available to train models, so a Python workflow was created to train our object recognizer model and semantic analyser model. The framework has further dependencies to other state-of-the-art machine learning libraries such as OpenCV (EMGU CV) and Accord.NET.

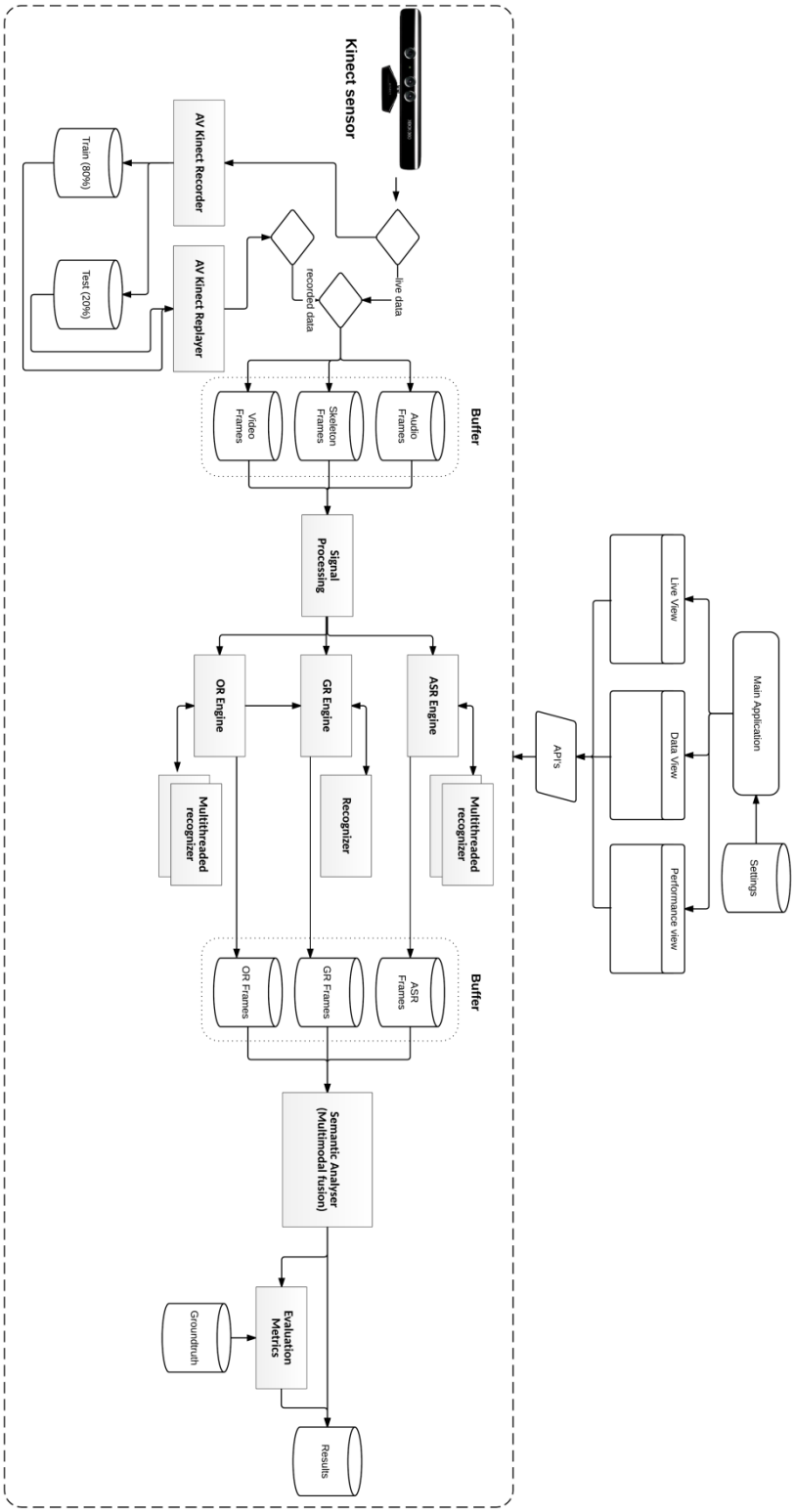


Fig 4-2 *Implementation of the multimodal ASU Framework*

4.3 Automatic Speech Recognition

Implementation

In chapter 2.1 we mentioned the components of modern ASR systems, specifically an acoustic and language model need to learned. We took an off-the-shelf acoustic model and speech engine (Microsoft Speech Platform v.11) for English US because it was optimized for the Kinect sensor. An additional benefit , due to the small vocabulary, was that the model didnt require a training phase per speaker. We couldn't confirm the used algorithm for the speech engine but we suspect a GMM-HMM variant. The recognized sequence words are provided with a confidence score and list of other utterance possibilities (with confidence score).

We experimented with a large vocabulary ("dictation mode") by implementing the system.speech API and routing it from to the Kinect. Unfortunately performance was very bad and required lenghty training of the acoustic model. With a small vocabulary and the above mentioned Kinect optimized acoustic model, we generated a language model consisting of 3 verbs (grab,push,pull) , 3 adjectives (red,blue,green) and 3 nouns (block, ball cup). Concatenating these in full utterances resulted in 27 variants. We splitted the training set in 80% training (20 particpants) and 20% test (5 participants), with each sample of the 213 test samples being modified with added Gaussian white noise before sending to the ASR.

Model evaluation

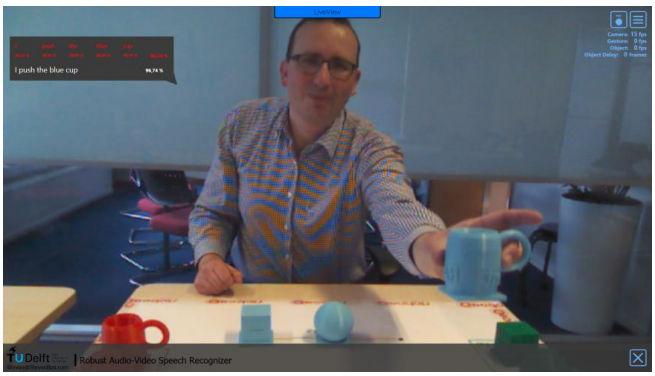


Fig 4-3 *An active ASR, showing the result in a speech bubble*

TABLE 4-1 ASR Baseline

SnR (dB)	WER (%)	Accuracy (%)
20	12.36	87.64
10	10.49	89.51
5	22.38	77.62
-5	95.62	4.38

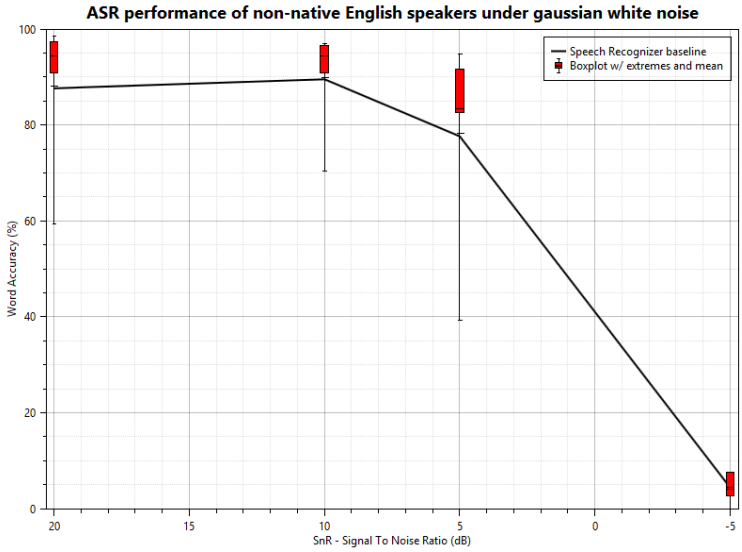


Fig 4-4 *ASR Results*

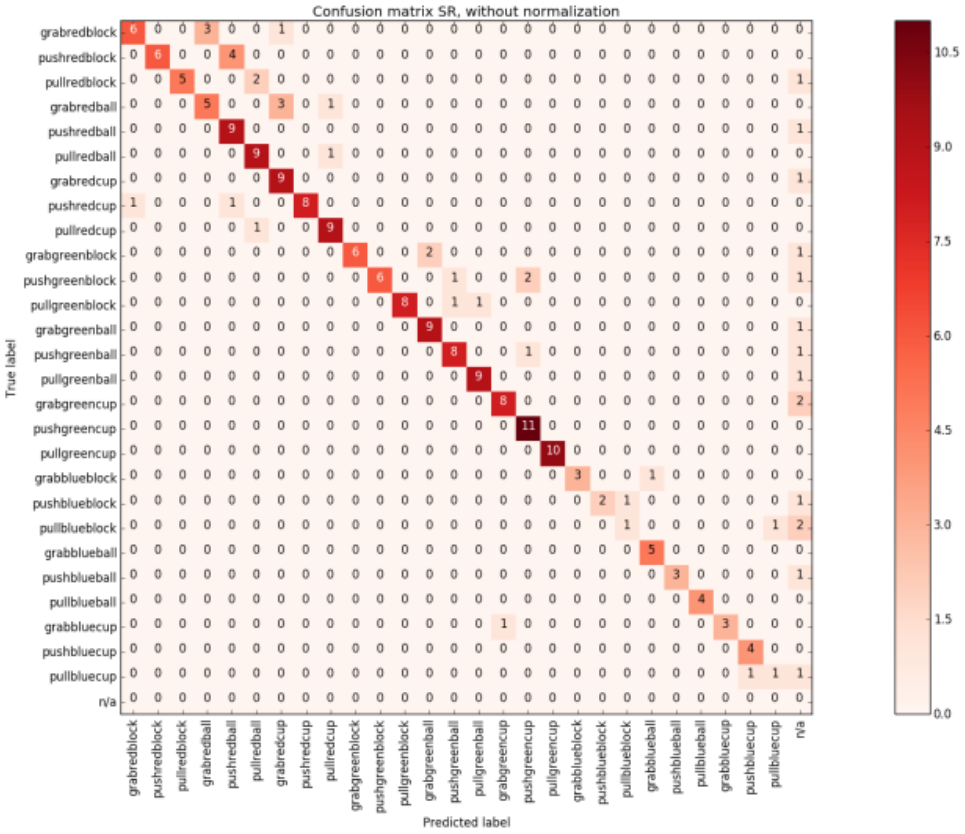


Fig 4-5 *Confusion matrix 20 dB*

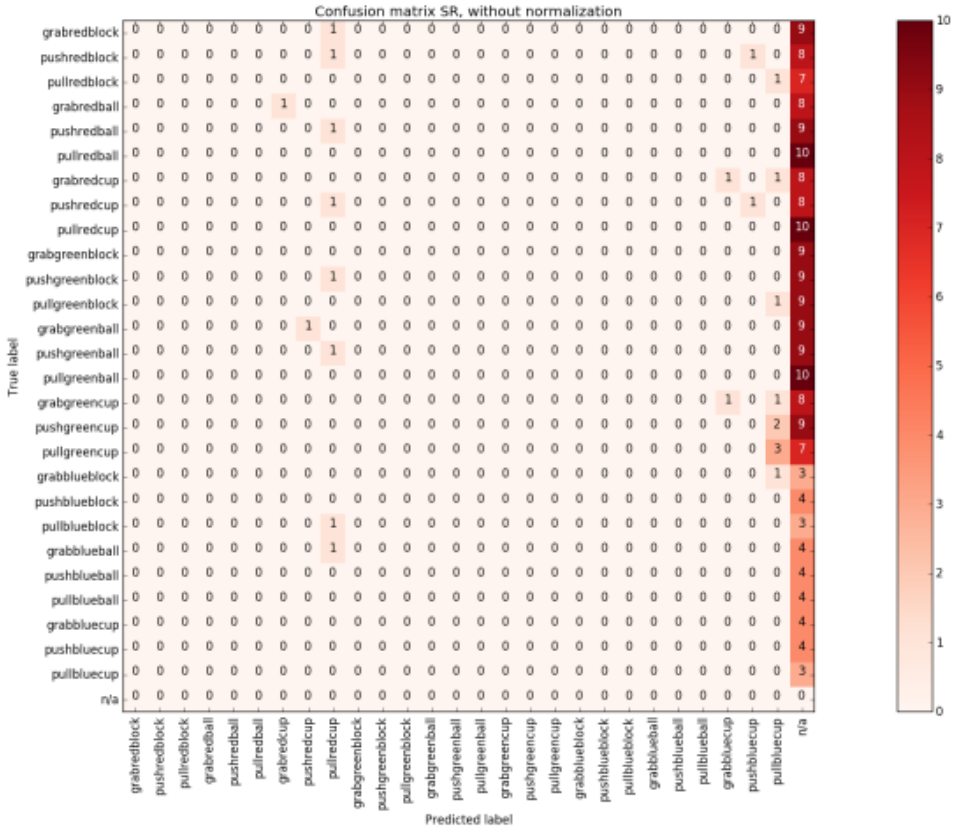


Fig 4-6 *Confusion matrix -5 dB*

4.4 Object Recognition

Implementation

We wrote a custom object recognizer pipeline (see fig. 4-7) that exploits depth information for both accurate detection and recognition of our data. With depth information, segmenting objects from the background becomes trivial. Also, with depth information normal vector calculations can be performed for accurate surface (eg. a table) and edge (the edges of an object) detection.

The detector algorithm is designed for our dataset and works in isolation, scanning the whole image for 3D blobs at 30 Hz. One of its features is to detect and track objects without prior marking (no supervision required). We use this feature to train the color models unsupervised.

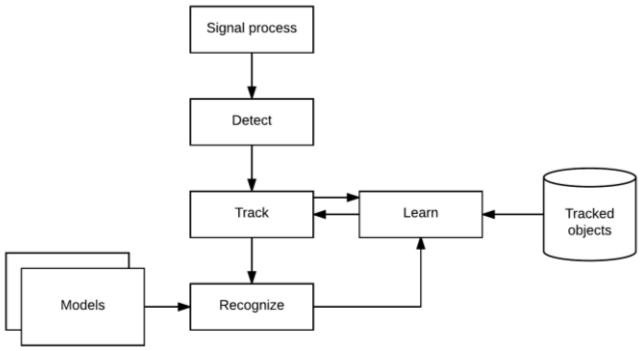


Fig 4-7 *Object recognizer pipeline*

Algorithm

Phase1: For each frame, initialize two canvasses, one all black for recognizer and one transparent for UI

Phase2: For each pixel check if within interaction volume, if yes paint RGB pixel on UI canvas and white pixel on recognizer canvas. Smoothen both RGB and Depth values on canvasses using a direct nearest neighbour search to fill 1pixel gaps that are inherent to the Kinect sensor.

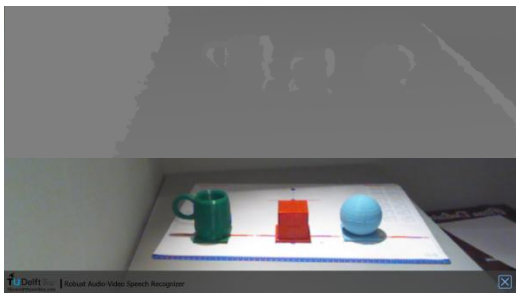


Fig 4-8 *Phase1: Input (color+depth)*

Fig 4-9 *Phase2: Depth Segmentation*

Phase3: Calculate the normal vector for each pixel and remove all upwards facing pixel (removing the table and segmenting the objects).

Phase4: Use the OpenCV erode function on the recognition canvas to create blobs. Detect the blobs using the OpenCV contour estimator.

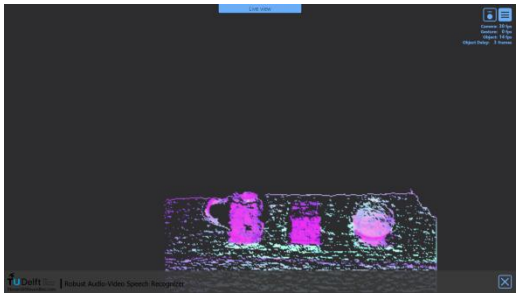


Fig 4-10 *Phase3: Floor removal*

Fig 4-11 *Phase4: Detection*

Phase 5: Foreach object candidate (contour) test for shape using three rule-based recognizers and determine color using a color histogram distance function.

Phase 6: Foreach recognized object add object to tracking memory and add to tracking history when position is < positionDeltaThreshold . When tracked multiple times change flag to "isTracked".

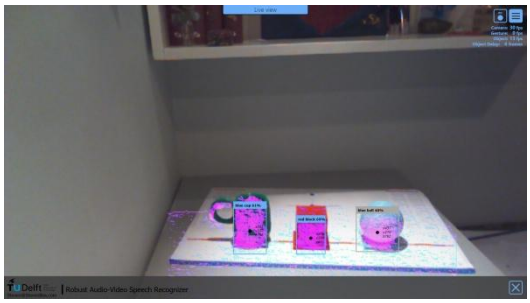


Fig 4-12 *Phase5: Recognition*

Fig 4-13 *Phase6: Tracking*

Model training

Color

For building a color model we used a 360 bin histogram representation and sampled each object for a total of 1000 frames and then normalized. We use the "histogram intersection" distance measure to compare two color histogram.

Shape

We used three trivial rule based recognizers to recognize the shapes:

- Blocks: object blob area < threshold (the block is the smallest object)
- Ball: circle approximation and distance measure towards the best matching circle
- Cup: height/width ratio as the cup has a larger height then width

Model evaluation

TABLE 4-2 OR Baseline

SnR(dB)	WER (%)	Accuracy (%)
n/a	26.06	73.94

TABLE 4-3 Precision/Recall

SnR(dB)	Precision (%)	Recall (%)
n/a	423/627= 67.46	423/633= 66.82

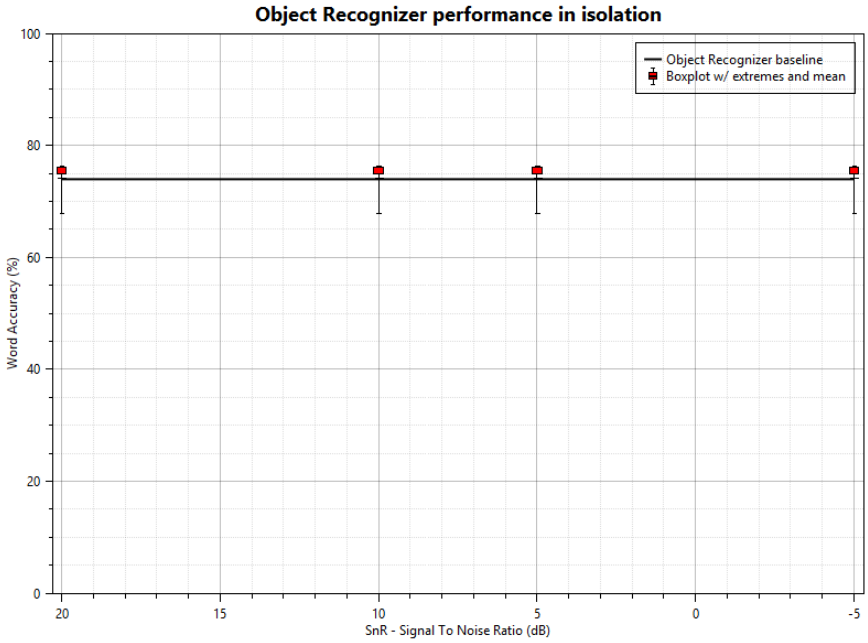


Fig 4-14 *OR baseline*

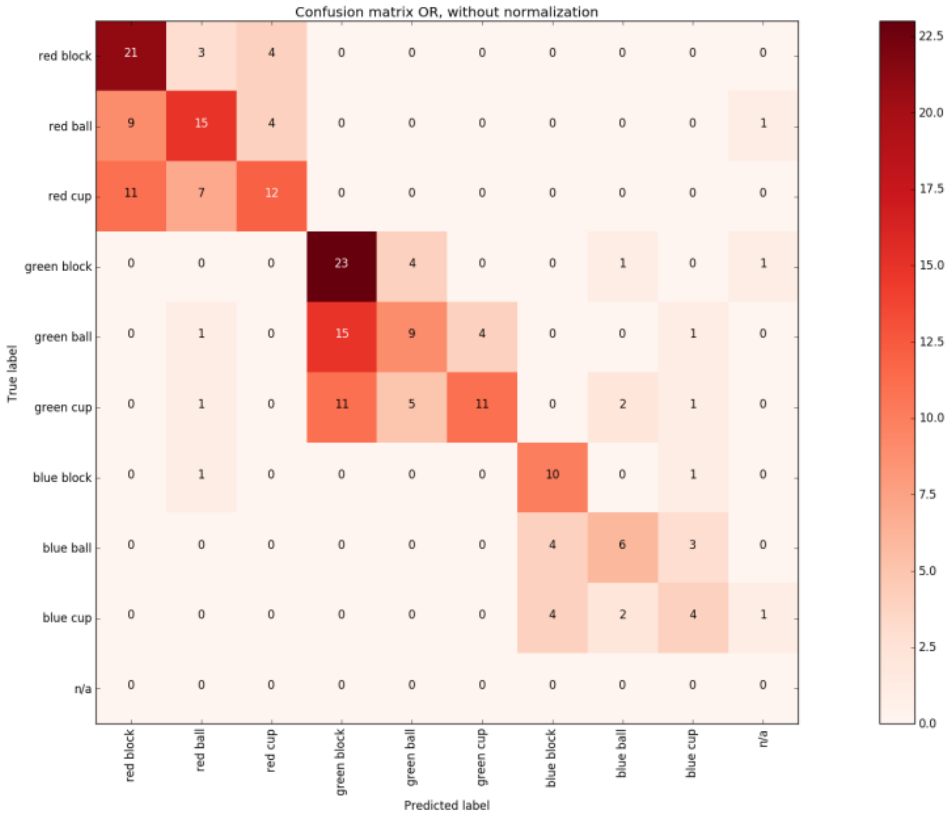


Fig 4-15 *Confusion matrix*

4.5 Gesture Recognition

Implementation

We wrote a custom gesture recognizer pipeline (see fig. 4-16) that exploits depth information to detect both object-hand interaction and recognize the type of gesture. In 2D, depth estimation by area size is common, however with 3D depth information, better predication (and thus tracking) is possible as the system can now predict an occluder and occludee. Our system only uses depth information to detect if and which hand of the participant interacts with a moving object.

The skeleton stream of the Kinect was very jittery. We used Holt Double Exponential Smoothing[8] to prevent jitter, but this did not prevent accurate readings when a hand was occluded by the object. The gesture recognizer performs its gesture detection algorithm (explained below) on a tracked object and uses the most frequently found object recognition results as the object of interest for a detected hand-object interaction.

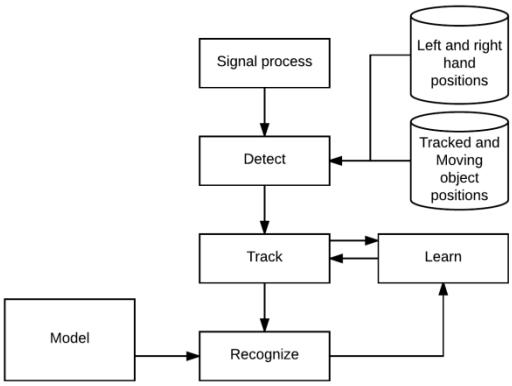


Fig 4-16 *Gesture recognition pipeline*

Algorithm

Phase0: Initialize two empty hands at start

Phase1: Foreach skeleton frame update hand position and keep history of positions

Phase2: Foreach tracked object detect if it moves by averaging the last 10 positions and check if the absolute sum is true of either (X > 5 cm and Y > 5 cm) or (X > 5 and Z > 5 cm) or (Y >5 and Z > 5)

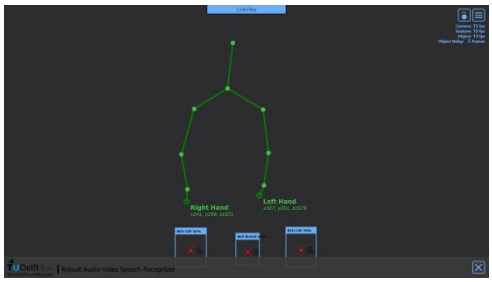


Fig 4-17 *Phase1: Input (skeleton+tracked objects)*

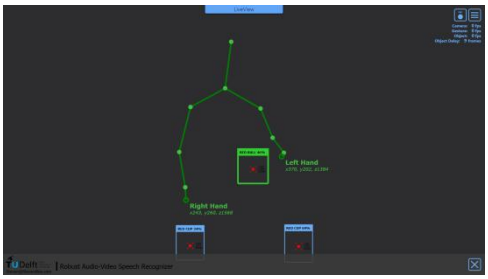


Fig 4-18 *Phase2: Detection of movement*

[8] http://msdn.microsoft.com/en-us/library/jj131024.aspx

Phase3: For each moving object and tracked object, detect if hand of participants correlates with object movement by computing the correlation in X, Y and Z direction of maximum 30 frames (minimum 5). We check correlation on the sign of the delta's (either positive or negative).

Phase4: For each correlated hand-object, recognize gesture using the model below by computing the position delta of the object away from the base position. We know the base position of Y and Z but not in X since some participants switch objects during the experiment). Compute final gesture result by averaging all gesture results and fire an event when correlation with hand-object correlation stops.

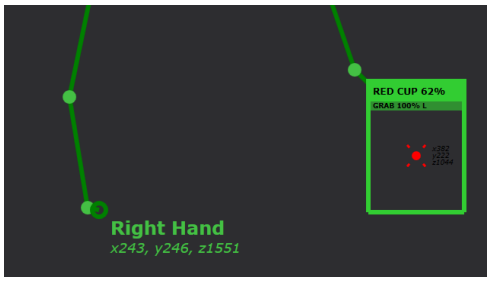


Fig 4-19 *Phase3: object-hand correlation*

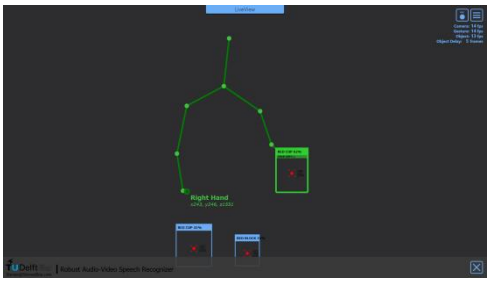


Fig 4-20 *Phase4: Recognition*

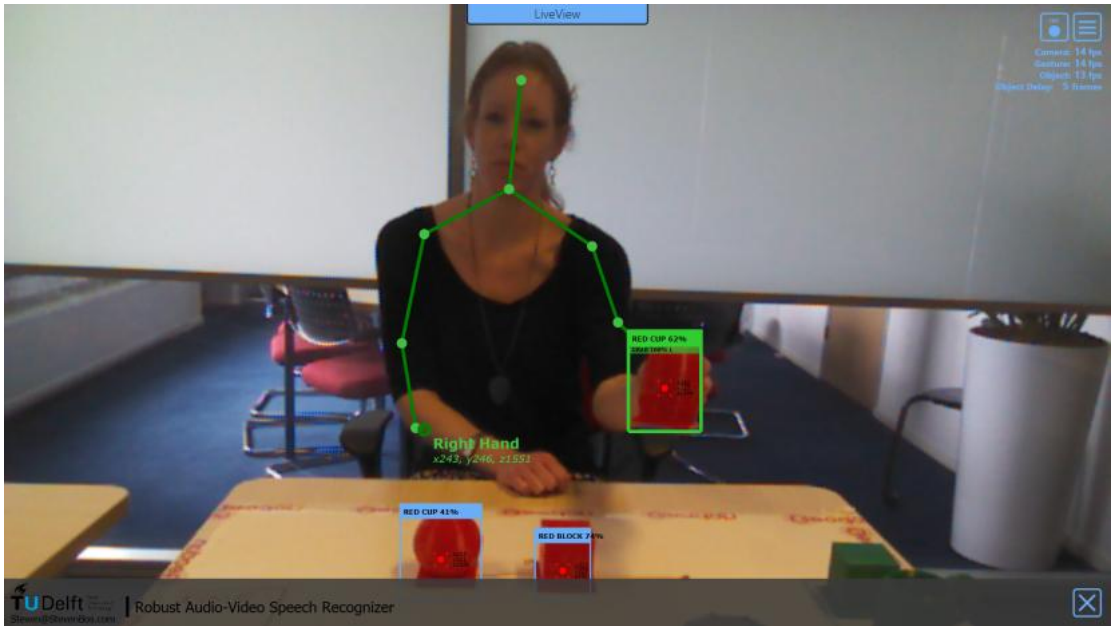


Fig 4-21 *Final result*

Model training

Gesture
For building a gesture model we used a trivial rule based detector (see fig 4-22).

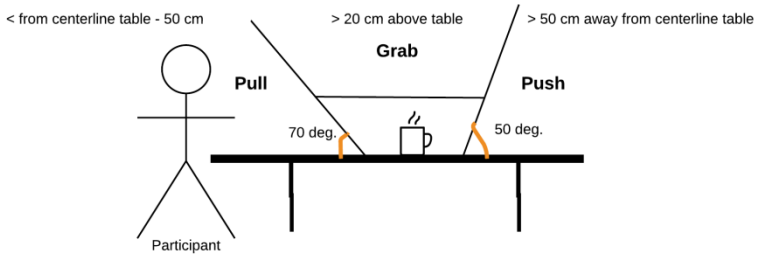


Fig 4-22*Rule based gesture detector with 2 angles*

Model evaluation

TABLE 4-4 GR Baseline

SnR(dB)	WER (%)	Accuracy (%)
n/a	43.31	58.69

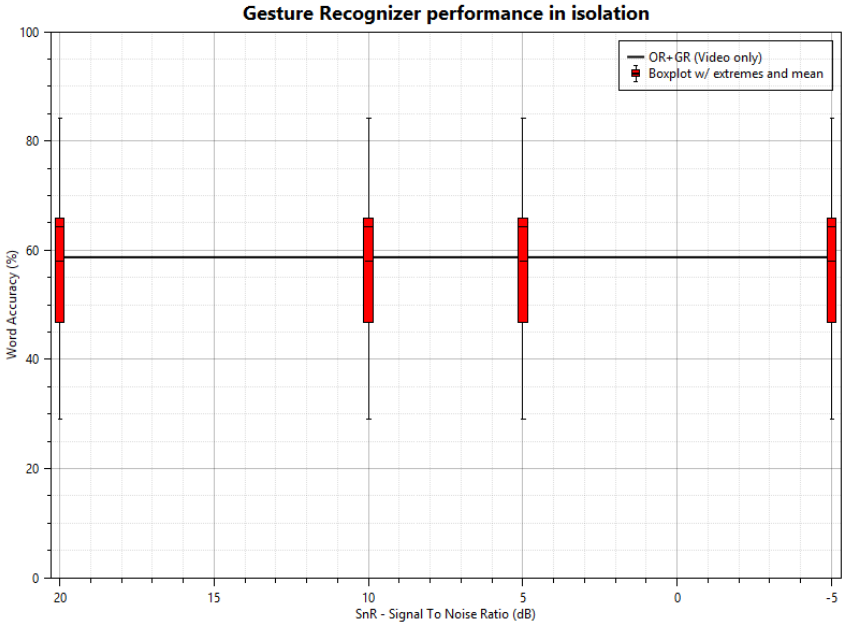


Fig 4-23 *GR baseline*

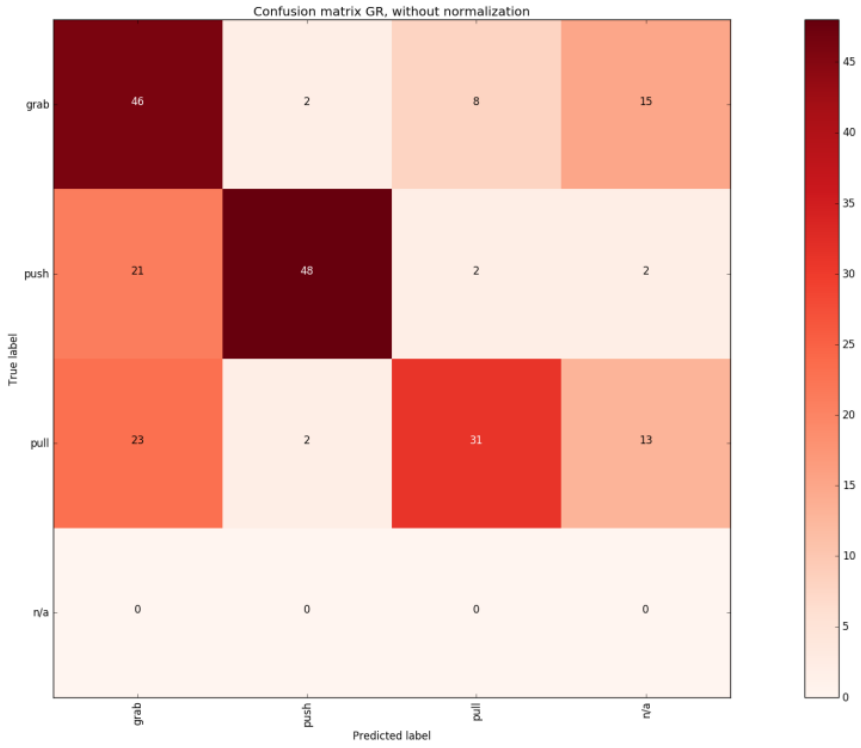


Fig 4-24*Confusion matrix GR*

TABLE 4-5 GR+OR Baseline

SnR(dB)	WER (%)	Accuracy (%)
n/a	36.62	63.38

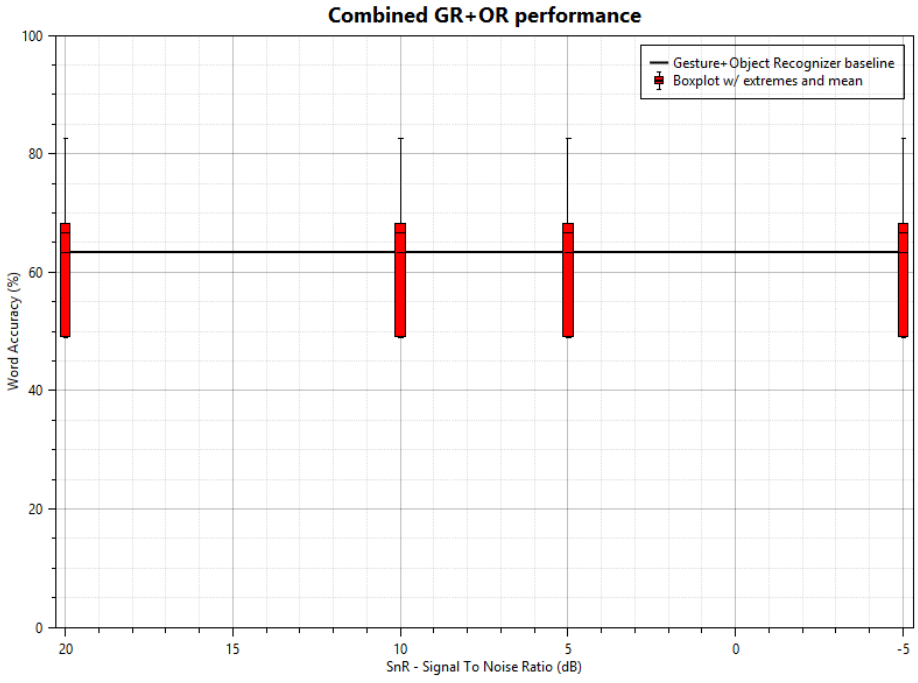


Fig 4-25 *GR+OR baseline*

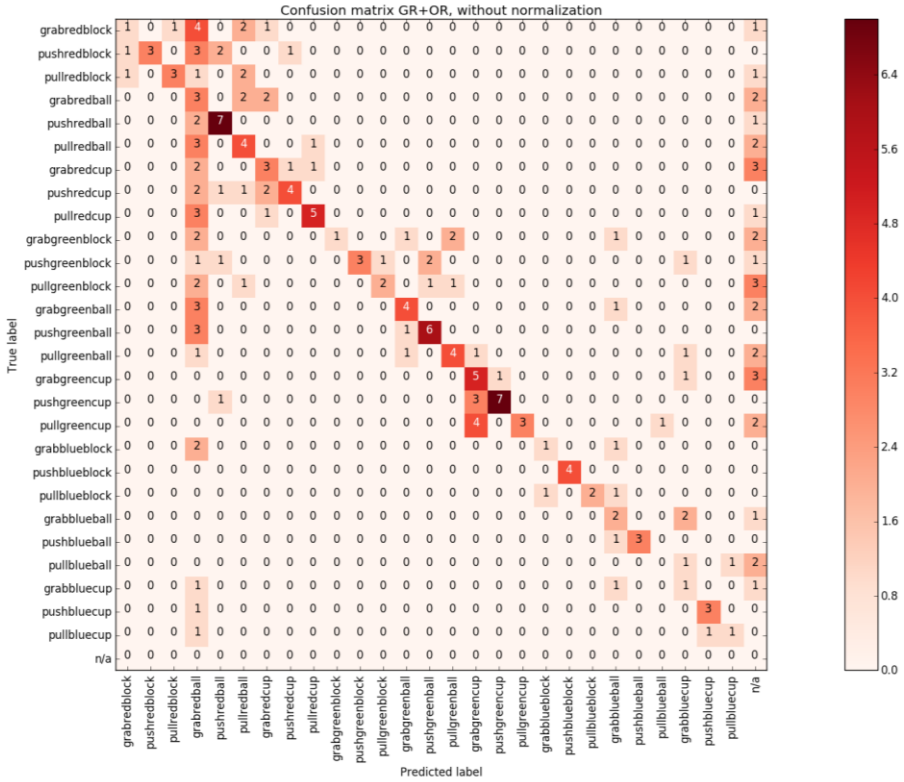


Fig 4-26 *Confusion matrix OR+GR*

4.6 Deep Neural Network Fusion

Implementation

We used the CNTK framework to train our deep neural network. We prepared our input vector in multi label one hot vectors (e.g. grab red block = [1 0 0 1 0 0 1 0 0]). We trained our model in Python and evaluated the WER performance in the MASU framework (c#) as training with CNTK is not yet available for C#.

Model training

We trained our model in 80%-20% fasion, with 3890 samples and tested the result with 852 unseen samples. We created an architecture as seen in fig 4-27 and used the following parameters:

- Randomized input for training
- Loss function: Binary cross entropy
- Error function: average epoch error function of all values (the error rate)
- Learning rate w/ momentum: 0.0003
- Learner: ADAM SGD
- Activation function Hidden layer 1: Leaky ReLU
- Activation function Hidden layer 2: Sigmoid

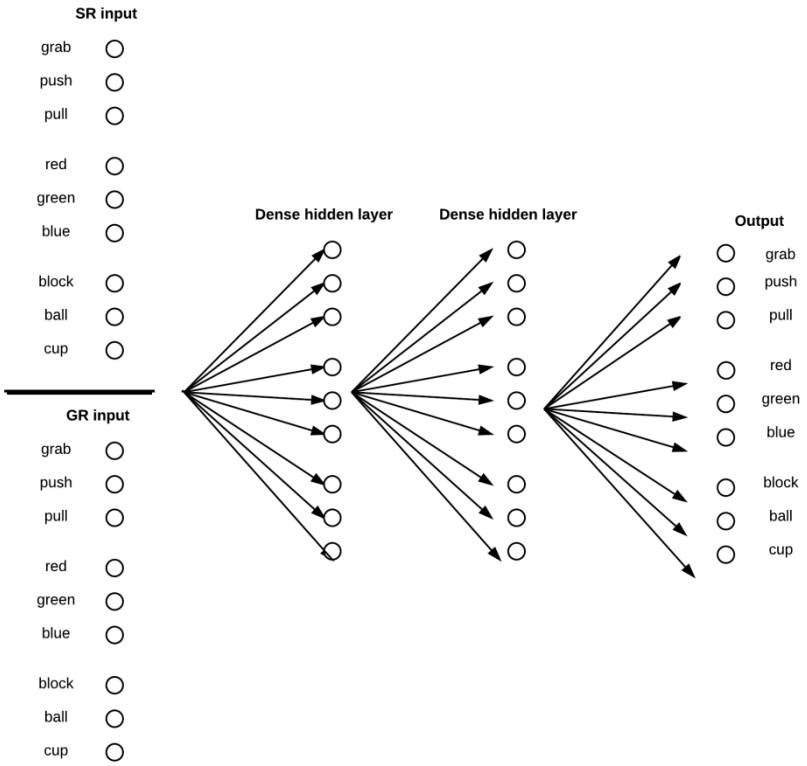


Fig 4-27 Neural network architecture

This resulted in the training plot in Fig 4-28, showing a clear convergence in both loss function and error.

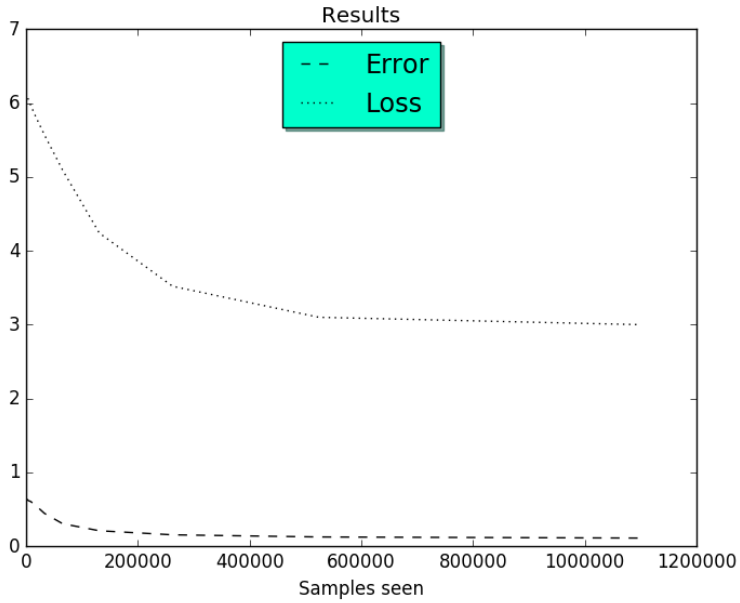


Fig 4-28 *Training plot*

Model evaluation

TABLE 4-6 *SR+GR+OR+DNN*

SnR (dB)	WER (%)	Accuracy (%)
20	7.67	92.33
10	7.52	92.48
5	14.87	85.13
-5	40.69	59,31

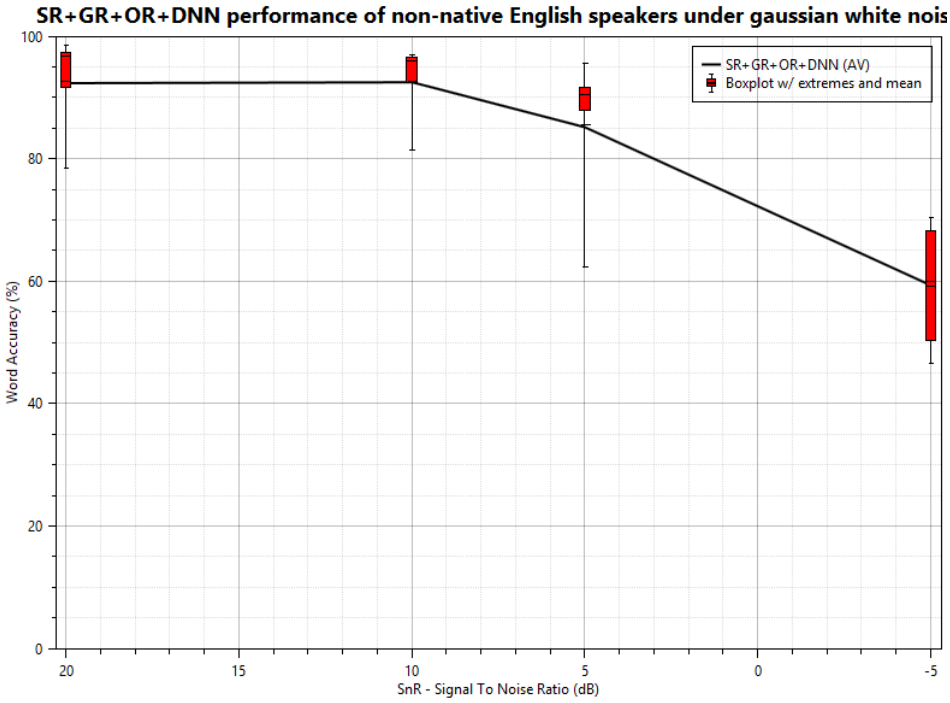


Fig 4-29 *SR+OR+GR+DNN performance*

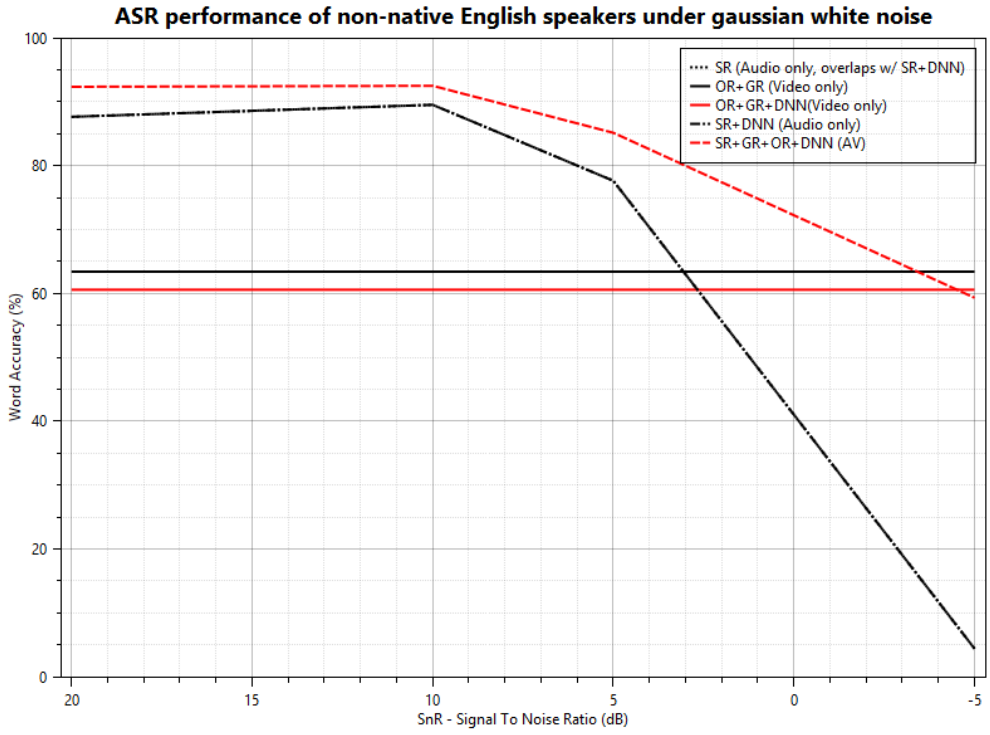


Fig 4-30 *Final overview of all recognizers*

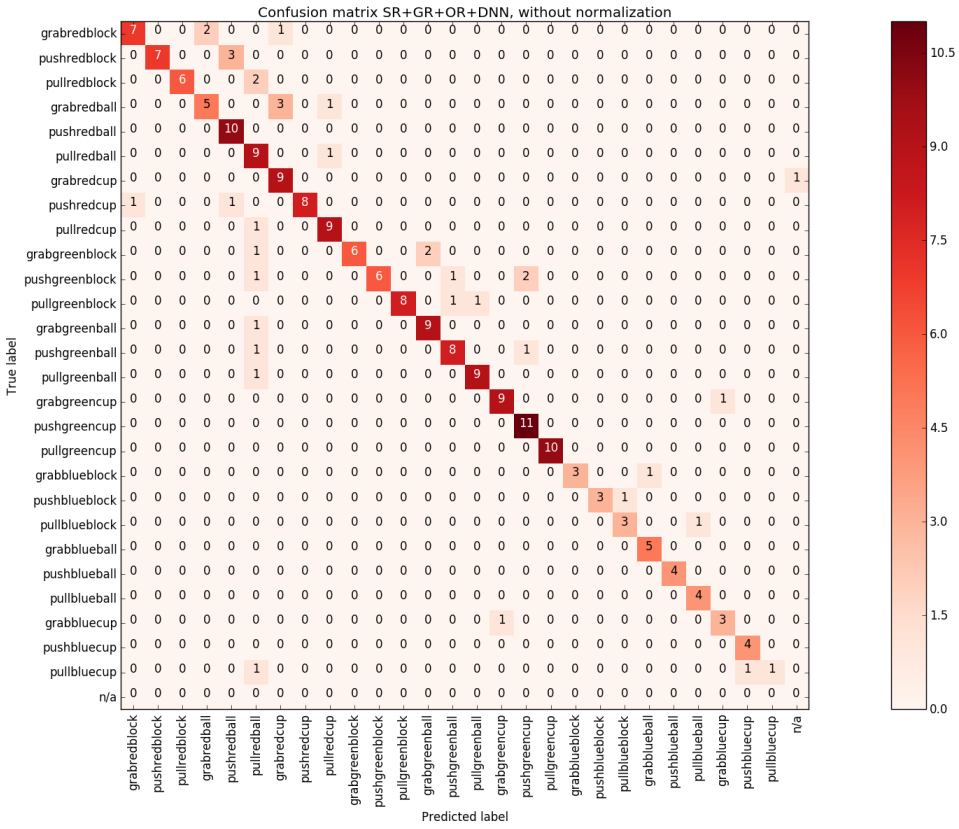


Fig 4-31 *Confusion matrix SR+GR+OR+DNN at 20 dB SnR*

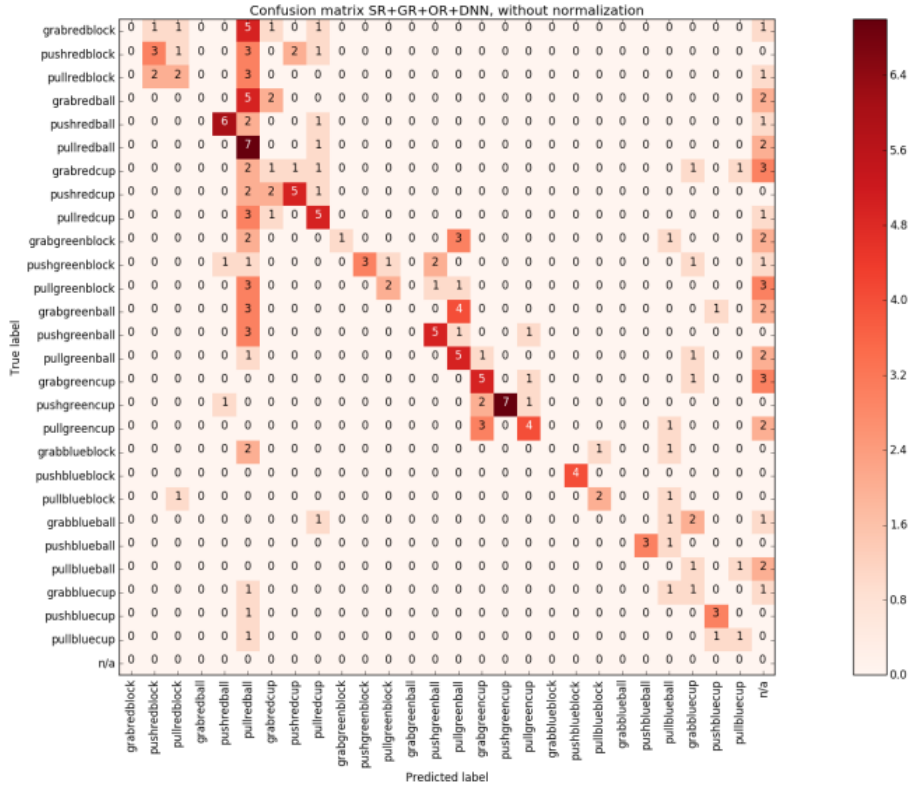


Fig 4-32 *Confusion matrix SR+GR+OR+DNN at -5 dB SnR*

TABLE 4-7 SR+GR+OR+DNN with ideal detector

SnR (dB)	WER (%)	Accuracy (%)
20	7.36	92.64
10	6.89	93.11
5	14.24	85.76
-5	34.43	65.57

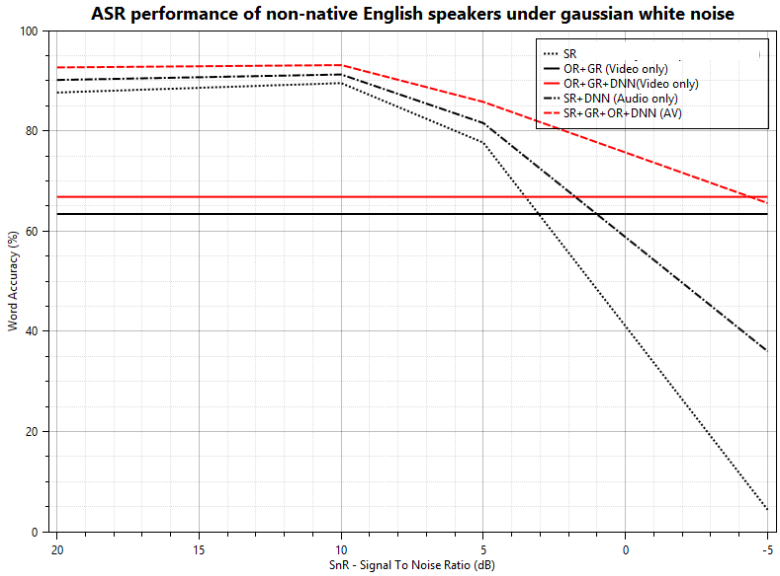


Fig 4-33 *Confusion matrix SR+GR+OR+DNN with ideal detector*

4.7 Discussion

Data

We discarded about 2% of our training data (82 of 3972 samples) as no event was generated by either speech nor gesture recognizer, which resulted in all zero, one hot vectors. We deliberately choose not to make both recognizers produce false positives (eg output events every second) to mimic a real situation. We did not discard the 4% (34/852 samples) of our test data. This means that our baselines will always underperform by at least 4% compared to an ideal event detector. We show in fig 4-31 that it really is worth improving our detectors. Another solution would be to include more datasources.

All session were annotated with a start and end mark, but sometimes due to frame lag, judging the start and end mark was difficult. In several occasions the particpants made error in the execution of the task, resulting in a total of 2% (29/1235) discarded groundtruth labels.

Automatic Speech Recognition

We generated Gausian white noise over the entire signal instead of only the speech part of the signal. This means that the signal-to-noise ratio is calculated on a part where there is a mix of speech and silence, which result in speech getting a lower noise as it is distributed.

The ASR occasionally generated false positives. These were not calculated in the WER score although one could interpret these false positives as 3 deletions (as there are always three words). We did not implement a weight for each word category. Weighing words makes sense for practical systems, where some words might have more importance.

The confusion matrices show that at low SnR levels few speech events are generated, contributing to the majority of the WER error. Humans have the ability to recognize speech under these noisy conditions with audio input only, confirming the large gap in Robust ASR performance in literature, e.g. (Delcroix & Watanabe, 2016).

Object Recognition

The OR consists of two recognizers for shape and color. Although the color recognizer approached near 100% precision, it is not light invariant (e.g. shadow, other light sources), making it unusable for practical systems. The shape recognizers performed poorly, as can be seen in the decision matrix and was optimized for the data set. A better solution would be to train a convolutional neural network which are proven to be good at this task. The benefit of this system was performance, as it is able to recognize and track at 30 Hz (the maximum of the Kinect v1) which is crucial for the detector of the gesture recognizer. This performance was not possbile with the current DNN implementation.

The experiment was designed such that the particpants did not occlude the objects too much, however in practice occlusion is very common. We did prevent a lot of sensor noise by preprocessing the signal extensively using both depth and color smoothing filters. The Kinect streams have many visual artifacts that prevent clean blob detection as well as variations in depth readings (average of 1.5cm with occasional extremes of 6cm were measured). As the Kinect v1 heavily depends on infrared light, using the system near windows in broad daylight or outdoors is not possible.

Gesture Recognition

The experiment requested to perform the gestures as natural as possible. Interestingly, some particpants used both hands to perfrom the task, while others did not. The duration of the task ranged from 1.4 seconds to approximately 6 seconds, especially for the first gesture of the experiment. The quality of the skeleton data from the Kinect was quite poor, especially around the extremities (hands). Although the Kinect support inferred and recognized joints, both were jittery and resulted in unstable depth readings. This problem increased when interacting with objects as the hand pixels were further minimized. The smoothing filter did prevent some jitter but was not nearly enough. There are also some interesting undocumented implementation "gotcha's" in the Kinect SDK like a reversed label for left and right hand, and cloned data when no new skeleton data is available in time.

The correlation between object and hand was hard due to the unstable behavior, resulting in poor gesture recognition and tracking performance. The confusion matrix show that the chosen implemention of the recognition algorithm also performed below average. Recurrent Neural Networks excel in finding non-linear decision boundaries in sequential data such as this, making it interesting to investigate. Gesture recognition might be just as hard as speech recognition as the variations are similarly large.

Multimodal fusion with DNN

Finding the right parameters, architecture and components is the magic part in training neural network. With CNTK playing and testing various options is easy although analysing results proved cumbersome. The week before this thesis was due CNTK released support for TensorBoard making visual analysis possible, a crucial time-saving tool. Our network converged fast, with a limit of about 11% error in training and 7% in testing.

Part of this limit is due to the chosen error function which was not identical to an edit distance metric such as WER. Suprisingly, the network already trained towards our target result, hinting for improvement when we do choose for an edit distance error function.

We missed training our network in C#, as some work needed be done double such as data preparation and data visualisation. Training in Python was fast though and in combination with GPU-offloading, implementing a DNN for a practical system was a breeze.

5

Conclusions

This chapter revisits our research question and working hypothesis. We will present our findings using a deductive reasoning approach, discuss limitations on the current work and implications for possible future directions.

The research question posed in the introduction was as follows:

- Can multimodal fusion of stereo audio and 3D video improve ASR performance?

We made this question more specific and measurable with our working hypothesis:

- Fusing classified objects and gestures from a 3D camera with speech have a beneficial effect on the WER performance of an Automatic Speech Recognizer under increasing Gaussian white noise condition from 20 dB up to -5 dB SnR compared to speech only.

Our results confirm that multimodal fusion of classified object, gestures and speech results under various Gaussian white noise conditions increase ASR performance significantly.The benefit of multimodal fusion using a deep neural network compared to both audio and video baselines is +5% accuracy at 20 dB SnR up to +61% accuracy at -5 dB SnR. The system uses the deep neural network to fuse classifer results, weighing results per word to reach a final outcome. It outperforms the naïve approach of selecting a whole modality above a certain threshold. Early results show that if detection rate is improved of the ASR component, the ASU system could benefit strongly from scenarios with audio only input (eg. when it is dark or with occlusion) as it can use the learned fused distribution.

5.1 Contributions

5.1.1 Multimodal Automatic Speech Understanding (MASU) framework

We presented a validated workflow to record, analyse and playback multimodal data, the MASU framework. The framework uses API's to communicate with the framework such that we could swap and test various recognizers. The multi-threaded design and performance of the framework enables data processing in real-time, with two buffers to deal with asynchronous nature of multimodal data and unpredictable availability of classifer output. This allows for building multimodal fusion application for early, interactive and late fusion strategies.

The framework is programmed in C#, potentially limiting its performance compared to unmanaged languages such as C++. Also, the C# wrapped library of the popular OpenCV is not robust, making debugging troublesome. The neural network library CNTK currently only supports CPU and GPU based model evaluation only. We used Python to train our neural network.

5.1.2 Annotated multimodal AV Corpus for 3D scene and intent recognition

Synchronous multimodal corpora are rare, especially speech in a 3D scene with actor-object interactions. There is a clear need for these datasets as deep learning algorithms and multimodal fusion algorithms in general (curse of dimensionality) requires incredible amounts of data to train. The multimodal corpus was recorded with a Kinect camera and microphone array and offers synchronous speech, depth and 3D skeleton data. According to the classification scheme in (Firman, 2016) it is rated:

- Realism: Type 2 (out of 3). Real environment and objects but scene was simplified for experiment
- 3D Completeness: Type 1 (out of 5). Only one 3D camera with single point of view (resulting in 2,5D data).

The labelled dataset of 25 particpants, 1206 samples per noise level is available for the research community and contains limited miscellaneous data such as pointing, stacking and contradicting speech-gesture intentions.

5.1.3 Algorithm for 3D hand-object detection and tracking

We describe an algorithm for 3D hand-object detection and tracking exploiting the depth sensing capabilities of 3D sensors to segment objects from the background and use the temporal motion vector correlation between actor hand and moving object to determine object which is controlled by the actor.

The algorithm, especially the detector part, works good for this experiment but is still under development for application on other datasets. The algorithm depends on quality skeleton data, which the Kinect with minimal signal processing cannot really deliver. Especially when occlusion occurs as the actor's hand wraps the object. Tracking is also lost when occlusion occurs for extended duration as the current object recognizer assumes that objects appear in roughly the same neighbour as where it dissappeard. This problem is mitigated when using large objects such that the object recognizers have at least some bits to continously track. Future versions of the algorithmn will experiment with Kinect 2 skeleton data and better object recognizers for handling occlusion and tracking problems.

5.1.4 Recipe to train deep neural network for multimodal fusion

To train the multimodal fusion classifer we used a late fusion strategy with a non-weighted concatination of classifier output. Each classifier outputs in onehot vector format, which after concatination becomes a multi label input (and output). We played with various settings using the CNTK framework such as various activation function (eg. leaky ReLU), various SGD approaches (eg. ADAM), various training settings and patterns (eg. high learning rate at first epochs, low later). We managed to train a two (dense) layer DNN with 20% of our dataset, with a total of only 852 samples (213 per noise level), which is considered few for deep neural networks. When samples have been seen 1.000.000 times by the network , the network stabilizes at around 11% training error and 7% test error. Interestingly enough,

further decrease of both training and test error (cross-entropy with softmax metric) increases the actual test error (WER metric). We didn't investigate this further.

5.1.5 Improved Robust AV Automatic Speech Recognizer using Deep Neural Network

We used an off-the-shelf speech recognizer, optimized for the Kinect sensor properties such that the microphone array could reliably recognize from a small distance. The constructed language model of 27 utterances can be consided small compared to the state-of-the-art large vocabulary ASRs, which made recognition much easier. We used an American-English acoustic model which worked well with the Dutch-English speaking particpants when background noise was little but degraded quickly when the signal reached below 5 dB SnR.

The largest problem with our ASR was its ability to detect speech when noise was added, resulting in most of the performance degradation, the quality of prediction was second by a large distance. We also measured the confidence values of our ASR system and noted that these drop when noise increases. We will test cross-particpant reliability in future research as adding confidence values should help the DNN in its decision to fuse modalities. A better ASR with better speech detection based on confidence values can teach the semantic analyser a limited form of Socrates' Wisdom, as it will know when too few evidence was found to emit a likely answer.

Improving our speech recognizer by introducing more modalities and using a deep neural networks worked really well for our data set, confirming our biologically inspired intuition how human process increase understanding with more evidence (data) and with more data sources. The proof-of-concept is currently not be able to deal with contradicting speech-gesture input or with data outside of the learned domain. For this technology to be adopted in practical scenario's, most of the work needs to be done in engineering better ASR and GR detection. Since we use the WER metric, performance in penalized per word and the DNN is able to realibly recognize objects from the OR stream, which is 66% of the result in our experiment.

5.2 Recommendations for Future work

During the research and development of the MASU framework and POC, a series of research directions arose which were cut short due to time constraints, they are described below. We also faced quite a few impediments that resulted in little time to optimize training the network so we are hopeful to squeeze out more performance in the near future. The main and pragmatic focus remains solving the cocktail problem using an embodied, usage-based approach while at the same time simplify prototyping ASU applications.

We trained the semantic analyser using a late fusion strategy. Related work described in chapter 2 show that early fusion is also a viable strategy. This way, the system can learn unsupervised or semisupervised which features to track from raw data and learn patterns, correlations and causalities between modalities that are otherwise lost when only considering confidence values.

For this thesis three modalities were chosen. Learning richer joint audio-visual representations would enable a host of new ASU applications. With face tracking we can visually distinguish which person is talking to whom and with lip reading we can aid the

speech recognizers as is done in mentioned literature. Mouth tracking would further enable a better estimate (detector) when speech starts and ends, predict speech behavior (eg. children, elderly) and possibly esitmate emotions from both speech and face/mouth.

The structure of the joint multimodal (knowledge) representation is a topic of heavy research. In this thesis we use the connectionist approach by using a deep neural network to model knowledge. Current limitations are sharing knowledge, extending knowledge across domains and the black box nature of neural networks. It would be interesting to use other structures to capture patterns as described in (Bos, 2012). We are keen on implementing and testing more principles of the Neural Theory of Language (Feldman J. , 2006) such as Embodied Construction Grammars (Bergen & Chang, 2003).

Humans are great at reading intentions when speech and body language contradict and selecting the most probable one given the prior behavior of the speaker, context and other factors. The humans brain uses inference and attention mechanisms to evaluate the weights of each input modality. To better deal with contradicting intents, our future system needs to track the physical context and speaker behavior through time and form a dialogue structure. This dialogue can be queried for hypothesis and truth assesment such as in IBM Watsons architecture which enables a form of inference.

During the development we demonstrated the use and performace of the framework with a POC - a limited language model, limited data and limited tools to profile results and models such as the neural networks. Scaling up the experiment with more data, such as more vantage points, (Firman, 2016) 3D completeness: type 2 or 3) and more actor-object interactions. More data and better tooling will directly and indirectly lead to increased WER performance.

Key part of this thesis was testing the robustness of the system under Gaussian white noise. Practical usability of the system involves robustness to more adverse conditions such as non-stationary white noise, audio compression, range and many types of low quality microphones. In this thesis we only tested audio noise, but robustness against various types of video noise such as occlusion, video compression, motion blur and bad lighting would make the system leaps more interesting. Finally, other types of robustness of the system would be interesting to investigate such as dynamic network topologies to temporarily filter noise in a particular situation. ResNets (deep residual neural networks) (He, Zhang, Ren, & Sun, 2016) are a new form of deep neural networks that feature "skip nodes" that allow removing (or disabling) whole hidden layers with only a few percentage of performance loss. We hypothesize that an interactive fusion strategy, ie. dynamically disabling layers that contain features that are sensitive to certain noise could actually increase performance.

Beyond Automatic Speech Recognition, towards Automatic Speech Understanding

Progress along these directions would bring us a few steps closer towards a grander vision with wide social and commercial impact: *Affective, Robust Automatic Speech Understanding (*ASU: any ASU)*.

- Affective. Early *ASU systems will not be flawless in its speech-to-semantic-text conversion, but to get social acceptance its output should be predicatable and correctable such that identical mistakes are rare - similar to current word completion solutions. User profiling with emotion recognition, contextual awareness and personalisation options with possibly emotive personalities are all required for the system to be practical in many situations and forgivable when mistakes happen (Karray, Alemzadeh, Abou Saleh, & Nours Arab, 2008).

.

- Robust. *ASU systems should achieve a level of performance that at least equals the human gold standard as use cases can be diverse and unpredictable. This means that we require to solve various speech signal degradation issues with larger distances (Delcroix & Watanabe, 2016) and (Barker, Vincent, Ma, Christensen, & Green, 2012) and others[9], multi-speaker background noise and compression artifacts such as in VOIP solutions. Also, as much research is focussed on the English language spoken by first-language users, more research in (international) dialects and child (directed) speech is required for it to be internationally viable.
- ASU. Understanding words beyond lexical and syntactic structures (symbols and grammar) requires learning word origins (etymology) - where and when are they used and why. This requires new grounded learning approaches with more modalities as words are always learned in some physical, emotional and affording context. Grounded learning in turn requires deeper multimodal fusion as words have limited meaning in a single modality but vast richness when combined (see table 2.2.1 six levels of cooperation). Finally, to truly understand language goes beyond training data and requires creative generation of new forms of usage - but within the limits of a language community. Models for language generation (eg. names, valid poetry) and common sense as well as capabilities for (biologically-inspired) inference are still areas of active research.

[9] http://www.cs.cmu.edu/afs/cs/user/robust/www/papers.html

Bibliography

Agarwal, A., Akchurin, E., Basoglu, C., Chen, G., Cyphers, S., Droppo, J., et al. (2014). An Introduction to Computational Networks and the Computational Network Toolkit. *https://www.microsoft.com/en-us/research/wp-content/uploads/2014/08/CNTKBook-20160217.pdf* , 1-150.

Aizenberg, I., Aizenberg, N., & Vandewalle, J. (2000). *Multi-Valued and Universal binary Neurons.* Springer US.

al-Dosari, M. (2016). Unsupervised Anomaly Detection in Sequences Using Long Short Term Memory Recurrent Neural Networks. *A Thesis Submitted to the Graduate Faculty of George Mason University in Partial Fullfillment of The Requirements for the Degree of Master of Science Computational Science* , 1-98.

Allen, J. (1987). Natural Language Undertstanding. 1-574.

Arisoy, E., Sainath, T., Kingsbury, B., & Ramabhadran, B. (2012). Deep Neural Network Language Models. *NAACL-HLT 2012 Workshop: Will we ever replace the N-gram model? On the future of language modelling for HLT* (pp. 20-28). Montreal: Assocation for Computational Linguistics.

Assael, Y., Shillingford, B., Whiteson, S., & Freitas, N. (2017). LIPNET: END-TO-END SENTENCE-LEVEL LIPREADING. *Under review as a conference paper at ICLR 2017* , 1-13.

Atrey, P., Hossain, M., El Saddik, A., & Kankanhalli, M. (2010). Multimodal fusion for multimedia analysis: a survey. *Journal Multimedia Systems* , 345-379.

Bailey, D. (1997). *A Computational Model of Embodiment in the Acquisition of Action Verbs.* Berkeley: Computer Science Division, EECS Department, University of California.

Baker, J. (1975). The DRAGON system—An overview. *IEEE Transactions on Acoustics, Speech, and Signal Processing* , 24-29.

Barker, J., Vincent, E., Ma, N., Christensen, H., & Green, P. (2012). The PASCAL CHiME Speech Separation and Recognition Challenge. *Computer Speech and Language* , 1-23.

Bengio, Y., Courville, A., & Vincent, P. (2014). Representation Learning: A Review and New Perspectives. 1-30.

Bergen, B., & Chang, N. (2003). Embodied Construction Grammar in Simulation-Based Language Understanding. *https://www1.icsi.berkeley.edu/~nchang/pubs/ecg.pdf* , 1-30.

Bing-Qiang, H., Guang-Yi, C., & Min, G. (2005). REINFORCEMENT LEARNING NEURAL NETWORK TO THE PROBLEM OF AUTONOMOUS MOBILE ROBOT OBSTACLE AVOIDANCE. *2005 International Conference on Machine Learning and Cybernetics* , 85-89.

Bos, S. (2012). The Natural Language Understanding Problem. *http://www.stevenbos.com/dl/RA/NLUreviewV1.02.pdf* , 1-108.

Bourlard, H., & Morgan, N. (1994). *Connectionist Speech Recognition - A Hybrid Approach.* Kluwer Academic Publishers.

Bregler, C., & Konig, Y. (1994). "Eigenlips" for Robust Speech Recognition. *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Adelaide, Australia* .

Bryant, J. (2008). *Best-fit Constructional Analysis.* Berkeley: University of California.

Cambria, E., & White, B. (2014). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE* , 48-57.

Chang, N. (2008). *Constructing Grammar: a computational model of the emergence of early constructions.* Berkeley: Computer Science Division, University of California.
Chang, N., & Mok, E. (2006). A structured context model for grammar learning. *Proceedings of the 2006 international joint conference on Neural Networks (IJCNN).* Vancouver.
Chelba, C., Zhang, X., & Hall, K. (2015). Geo-location for Voice Search Language Modeling. *Interspeech 2015* , 1-5.
Chol Song, Y., & Kautz, H. (2012). A testbed for learning by demonstration from natural language and RGB-depth video. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence* , 2457-2458.
Coulom, R. (2002). Apprentissage par renforcement utilisant des reseaux de neurones, avec des applications au controle moteur. *These pur obtenir le grade de Docteur de L'INPG. Specialite: Sciences Cognitives* , 1-168.
Cun, Y. (2016). Predictive Learning. *NIPS* , 1-75.
Dai, W. (2016). *Acoustic Scene Recognition with Deep Learning.* Carnegie Mellon University.
Davis, K., Biddulph, R., & Balashek, S. (1952). Automatic Recognition of Spoken Digits. *Journal of Acoustical Society of America* , 627-642.
Delcroix, M., & Watanabe, S. (2016). Recent Advances in Distant Speech Recognition. 1-207.
Dyer, C., Kuncoro, A., Ballesteros, M., & Smith, N. (2016). Recurrent Neural Network Grammars. *5th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.*
Estellers, V., & Thiran, J. (2010). Overcoming Asynchrony in Audio-Visual Speech Recognition. *Proceedings of Multimedia Signal Processing Conference* , 1-6.
Feldman, J. (2006). *From molecule to metaphor.* Cambridge, M.A.: The MIT Press.
Feng, X., Zhang, Y., & Glass, J. (2014). Speech Feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition. *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)* , 1778-1782.
Ferreira Junior, J. (2013). Advances in Computational Science, Engineering and Information Technology. *Proceedings of the Third International Conference on Computational Science, Engineering and Information Technology* , 1-326.
Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A., et al. (2010). Building Watson: An overview of the DeepQA project. *Association for the Advancement of Artificial Intelligence* , 59-79.
Firman, M. (2016). RGBD Datasets: Past, Present and Future. *https://arxiv.org/pdf/1604.00999.pdf* , 1-13.
Fleischman, M., & Roy, D. (2008). Grounded Language Modeling for Automatic Speech Recognition of Sports Video . *Proceedings of ACL-08: HLT* , 121-129.
Fry, D., & Denes, P. (1959). The Design and Operation of the Mechanical Speech Recognizer at University College London. *Journal of British Institution of Radio Engineers* , 211-229.
Galatas, G., Potamianos, G., & Makedon, F. (2012). AUDIO-VISUAL SPEECH RECOGNITION INCORPORATING FACIAL DEPTH INFORMATION CAPTURED BY THE KINECT. *20th European Signal Processing Conference* , 2714-2717.

Gergen, S., Zeiler, S., Abdelaziz, A., Nickel, R., & Kolossa, D. (2016). Dynamic stream weighting for turbodecoding-based audiovisual ASR. *In interspeech* , 2135-2139.
Glass, J. (2007, November). *A brief introduction to speech recognition.* Opgeroepen op December 17, 2016, van http://www.cs.columbia.edu/~mcollins/6864/slides/asr.pdf
Gloror, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. *Proceedings of the 14th Internatnioal Conference on AI and Statistics* , 315-323.
Gorniak, P., & Roy, D. (2003). Understanding complex visually referring utterances. *Proc. of the HLT-NAACL 2003 Workshop on Learning Word Meaning from Non-linguistic Data* , 14-21.
Graves, A., Mohamed, A., & Hinton, G. (2013). Speech Recognition with Deep Recurrent Networks. *IEEE International Conference on Acoustics, Speech and Signal Processing* .
Gravier, G., Axelrod, S., Potamianos, G., & Neti, C. (2002). Maximum entropy and MCE based HMM stream weight estimation for audiovisual ASR. *IEEE International Conference on Acoustics, Speech, and Signal Processing* , 853-856.
Gravier, G., Potamianos, G., & Neti, C. (2007). Asynchrony Modeling for audio-visual speech recognition. 1-4.
Grifioni, P. (2009). Multimodal Human Computer Interaction and Pervasive Services. 538.
Hagoort, P. (2005). On Broca, brain, and binding: a new framework. *Trends in Cognitive Sciences* , 9(9): 416-423.
He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition* , 1-12.
Heck, L., Konig, Y., Kemal Sönmez, M., & Weintraub, M. (2000). Robustness to telephone handset distortion in speaker recognition by discriminative feature design. *Speech communication* , 181-192.
Hinton, G., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation* , 1527-1554.
Hochreiter, S. (1991). *Untersuchungen zu dynamische neuronalen Netzen.* Technische Universität München.
Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation* , 1735-1780.
Huang, X., & Deng, L. (2010). An overview of modern speech recognition. In *Handbook of Natural Language Processing* (pp. 339-366). Chapman&Hall.
Huang, X., Baker, J., & Reddy, R. (2014). A Historical Perspective of Speech Recognition. *Communications of the ACM* , 94-103.
Iroju, O., & Olaleke, J. (2015). A Systematic Review of Natural Language Processing in Healthcare. *Information Technology and Computer Science* , 44-50.
Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing* , 67-72.
Ivakhnenko, A., & Lapa, V. (1965). *Cybernetic Predicting Devices.* CCM Information Corporation.
Jelinek, F. (1976). Continuous speech recognition by statistical methods. *Proceedings of the IEEE 64* , 532-556.
Jelinek, F. (2009). The Dawn of Statistical ASR and MT. *Computational Linguistics* , 483-494.
Juang, B., & Rabiner, L. (2004, August 10). *Automatic Speech Recognition – A Brief History of the Technology.* Opgeroepen op December 28, 2016, van

http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LALI-ASRHistory-final-10-8.pdf
Jurafsky, D., & Martin, J. (2000). *Speech and Language Processing.* New Jersey: Prentice-Hall.
Karray, F., Alemzadeh, M., Abou Saleh, J., & Nours Arab, M. (2008). Human-Computer Interaction: Overview on State of the Art. *INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS, VOL. 1, NO. 1* , 139.
Knight, W. (2016). AI's Language Problem. *MIT Technology Review* .
LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature* , 436-444.
Li, J., Monroe, W., Ritter, A., Galley, M., Gao, J., & Jurafsky, D. (2016). Deep Reinforcement Learning for Dialogue Generation. 1-11.
Liberman, M. (2010). Obituary - Fred Jelinek. *Association for Computational Linguistics* .
MacCartney, B., & Potts, C. (2016). Natural language Understanding. *http://web.stanford.edu/class/cs224u/materials/cs224u-2016-intro.pdf* , 1-38.
Martin, T. (1970). *Acoustic recognition of a limited vocabulary in continuous speech.* Philadelphia: University of Pennsylvania.
Matuszek, C., Fitzgerald, N., Zettlemoyer, L., Bo, L., & Fox, D. (2012). A Joint Model of Language and Perception for Grounded Attribute Learning. *Computer Science and Engineering* , 1-8.
Mavridis, N., & Roy, D. (2006). Grounded Situation Models for Robots: Where words and percepts meet. *IEEE Conference on Intelligent Robots and Systems IROS 2006* , 4690-4697.
Mendis, C., Droppo, J., Maleki, S. M., Mytkowicz, T., & Zweig, G. (2016). Parallelizing WFST speech decoders. *Proceedings of IEEE ICASSP* , 5325-5329.
Minker, W., Bühler, D., & Dybkjær, L. (2005). Spoken Multimodal Human-Computer Dialogue in Mobile environments. 1-406.
Mohri, M., Pereira, F., & Riley, M. (2008). Speech Recognition with weighted finite-state transducers. In L. Rabiner, & F. Juang, *Handbook on Speech Processing and Speech Communication.* Springer: Heidelberg.
Mok, E. (2008). *Contextual Bootstrapping for Grammar Learning.* Berkeley: Computer Science Department, University of California.
Movellan, J., & Mineiro, P. (1998). Robust Sensor Fusion: Analysis and Application to Audio Visual Speech Recognition. *Machine Learning* , 32-85.
Mroueh, Y., Marcharet, E., & Goel, V. (2015). Deep multimodal learning for audio-visual speech recognition. *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSSP)* , 1-8.
Narayanan, S. (1997). *Embodiment in Language Understanding: Sensory-Motor Representations for Metaphoric Reasoning About Event Descriptions.* Berkeley: Computer Science Division, EECS Department, University of California.
Navigli, R. (2009). Word Sense Disambiguation: A survey. *ACM Computing Surveys* , Vol. 41, No. 2, Article 10.
Ng, A. (2011). Sparse autoencoder-CS294A Lecture Notes. *https://web.stanford.edu/class/archive/cs/cs294a/cs294a.1104/sparseAutoencoder.pdf* , 1-19.
Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. (2011). Multimodal Deep Learning. 1-8.
Nguyen-Duc-Thanh, N., Stonier, D., Lee, S., & Kim, D. (2011). A New Approach for Human-Robot Interaction Using Human Body Language. *Convergence and Hybrid Information Technology. ICHIT 2011* , 762-769.

Oquab, M., Bottou, L., & Laptev, I. (2014). Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. *CVPR* , 1-8.
Papandreou, G., Katsamanis, A., Pitsikalis, V., & Maragos, P. (2008). Adaptive Multimodal Fusion by Uncertainty Compensation With Application to Audiovisual Speech Recognition. *IEEE Transactions on Audio Speech and Language* , 1-13.
Phrasee.co. (2016). The Ballad of Tay- Microsoft's Spectacular Chatbot Fail. *https://phrasee.co/the-ballad-of-tay-microsofts-spetacular-chatbot-fail/* .
Potamianos, G., Neti, C., Gravier, G., Garg, A., & Senior, A. (2003). Recent Advances in the Automatic Recognition of Audio-Visual Speech. *Proceedings of the IEEE* , 1-17.
Reckman, H., Orkin, J., & Roy, D. (2010). Learning meanings of words and constructions, grounded in a virtual game. *Proc. of the 10th Conf. on Natural Language Processing (KONVENS)* , 1-9.
Reddy, R. (1976). Speech recognition by machine: A review. *Proceedings of the IEEE* , 501-531.
Regier, T. (1996). *The Human Semantic Potential: Spatial Language and Constrained Connectionism.* Cambridge, MA: The MIT Press.
Rosenblatt, F. (1957). *The Perceptron - a perceiving and recognizing automaton.* New York: Cornell Aeronautical Laboratory.
Rossiter, J. (2011). Multimodal intent recognition for natural human-robotic interaction. *A thesis submitted for the degree of Doctor of Philosophy-School of Electronic, Electrical and Computer Engineering* , 1-246.
Roy, D. (2003). Grounded Spoken Language Acquisition: Experiments in Word Learning. *IEEE: Transactions on Multimedia* , 197-209.
Roy, D. (2002). Learning Visually-Grounded Words and Syntax for a Scene Description Task. *Computer Speech and Language 16* , 1-39.
Roy, D., & Mukherjee, N. (2003). Visual Context Driven Semantic Priming of Speech Recognition and Understanding. *Computer Speech and Language* , 1-25.
Ruan, S., Wobbrock, J., Liou, K., Ng, A., & Landay, J. (2016). Speech Is 3x Faster than Typing for English and Mandarin Text Entry on Mobile Devices. *https://arxiv.org/abs/1608.07323* , 1-12.
Rumelhart, D., Hinton, G., & Williams, R. (1986). *Parallel Distributed Processing. Exploration of the Microstructure of Cognition vol.1: Foundations.* MIT Press.
Salakhutdinov, R., & Hinton, G. (2009). Semantic Hashing. *International Journal of Approximate Reasoning* , 969-978.
Saon, G., Sercu, T., Rennie, S., & Kuo, H. J. (2016). *The IBM 2016 English Converational Telephone Speech Recognition System.* Yorktown Heights, New York: IBM T.J. Watson Research Center.
Saunders, J., Lyon, C., Nehaniv, C., Dautenhahn, K., & Förster, F. (2008). Robot Learning of Holophrases, Words and Proto-Grammar from Simulated Babbling and Physical Interaction. *Proceedings of 2nd International IEEE Symposium on Artificial Life.* Nashville, Tennessee, USA.
Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks* , 85-117.
Schmidhuber, J. (1992). Learning complex, extended sequences using the principle of history compression. *Neural Computation* , 234-242.
Seide, F., Li, G., & Yu, D. (2011). Conversational Speech Transcription using Context-Dependent Deep Neural Networks. *Interspeech* , 437-440.

Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., et al. (2015). Accurate, Robust, and Flexible Real-time Hand Tracking. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* , 3633-3642.
Shotton, J., Kipman, A., Fitzgibbon, A., Finocchio, M., Moore, R., Blake, A., et al. (2011). Real-Time Human Pose Recognition in Parts from a Single Depth Image. *IEEE Computer Vision and Pattern Recognition (CVPR) 2011* , 1-25.
Song, Y., Kautz, H., Lee, R., & Luo, J. (2012). A General Framework for Recognizing Complex Events in Markov Logic. *https://www.cs.rochester.edu/u/kautz/papers/kautz-PAIR2013-general-framework.pdf* , 1-7.
Sparck Jones, K. (2001). Natural language processing: a historical review. *Artificial Intelligence Review* , 1-12.
Stanley, K., & Miikkulainen, R. (2002). Efficient Reinforcement Learning through Evolving Neural Network Topologies. *Proceedings of the Genetic and Evolutionary Computation Conference* , 1-9.
Sutskever, I., Vinyals, O., & Le, Q. (2014). Sequence to Sequence Learning with Neural Networks. *NIPS'14 Proceedings of the 27th International Conference on Neural Information Processing Systems* , 3104-3112.
Tamura, S., Iwano, K., & Furui, S. (2005). Toward robust multimodal speech recognition . 1-4.
Tebelskis, J. (1995). *Speech Recognition using Neural Networks.* Carnegie Mellon University.
Tomasello, M. (2008). The usage-based theory of language acquisition. *https://www.princeton.edu/~adele/LIN_106:_UCB_files/Tomasello-BavinChapter09.pdf* , 1-20.
Turing, A. (1950). Computing machinery and intelligence. *Mind* .
Veit, A., Wilber, M., & Belongie, S. (2016). *Residual networks behave like ensambles of relatievely shallow networks.* Cornell University.
Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* , 260-269.
Waibel, A., & Lee, K. F. (1990). *Readings in Speech Recognition.* Morgan Kaufmann.
Wechsung, I., Engelbrecht, K., Kühnel, C., Möller, S., & Weiss, B. (2012). Measuring the Quality of Service and Quality of Experience of multimodal human–machine interaction. *Journal on Multimodal User Interfaces* , 1-2.
Weizenbaum, J. (1966). ELIZA — A Computer Program for the Study of Natural Language Communication between Man and Machine. *Communications of the Association for Computing Machinery* , 9: 36-45.
White, G. M., & Neely, R. B. (1975). Speech recognition experiments with linear prediction, bandpass filtering, and dynamic programming. *Proceedings of 2nd USA-Japan Computer Conference.* Tokyo.
Winograd, T. (1975). Understanding Natural Language . *http://www.cs.uu.nl/docs/vakken/b3ii/Intelligente%20Interactie%20literatuur/College%208.%20Natuurlijke%20taal%20processing%20(Dignum)/Extra%20literatuur/shrdlu_Winograd.pdf* , 1-191.
Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., et al. (2016). *Achieving Human Parity in Conversational Speech Recognition.* Microsoft Research(MSR-TR-2016-71).

Ye, G., Liu, C., & Gong, Y. (2016). Geo-location dependent deep neural network acoustic model for speech recognition. *Acoustics, Speech and Signal Processing (ICASSP* , 5870-5874.
Zweig, G. (2016, October 18). *Blogs.microsoft.com*. Opgeroepen op December 2016, 28, van http://blogs.microsoft.com/next/2016/10/18/historic-achievement-microsoft-researchers-reach-human-parity-conversational-speech-recognition/#sm.0000g2h55w8wodscwdn2b88qbh74a