École Polytechnique Fédérale de Lausanne

Distributed System for Fast and Accurate Visual Speech Recognition

Master's Thesis in Robotics

Author: David Roch

Professor: Pierre Dillenbourg

Supervisors: Daniel Carnieto Tozadore

June 24, 2022

Abstract

The autonomy of social robots are not still on a level to robustly support teachers in classrooms. Using computer vision and neural networks, their perceptions of the environment have been greatly increased during the past years. However, there is still open challenges to make them better. One of the problem is that Audio Speech Recognition (ASR) algorithms are not precise enough, specially in noisy environment such as classrooms. By understanding better the children, the Human-Robot Interaction (HRI) can be greatly improved and so, having a more complex behaviour, activities and better autonomy. As most robots are already built with camera/s, the solution opted is lip reading. Its potential used alone is not high as there are too many visemes compared to the number of phonemes. Nevertheless, it has been shown that combining lip reading with ASR increase the accuracy in noisy environment. The difficulty here is to make the Visual Speech Recognition (VSR) algorithm run fast enough to be uses for HRI while keeping enough accuracy such to remain useful. Thus, we proposed a distributed system hierarchy where the robot (or a laptop connected to the robot) will record images and sounds while a server is dealing with building the videos and preprocessing it before doing the forward pass of the neural network to return a prediction of the users' speech. During the preprocessing, RetinaFace combined with the FAN Face Alignment algorithm was initially used for its precision to find face's landmarks but the results were slow. To speed up the process, we first built a distributed system able to gather data onboard while processing them on a server and then changed the Retina Fan algorithm for the BlazeFace network (MediaPipe). The resulting architecture was capable of classifying the sentences 63 times faster than classifying them on a laptop. We also collected data from English and non-English native speakers to validate the setup and evaluate the persistence of the results on non-English native speaker. Finally, a small variation in the classification performance was noticed when comparing English native speakers to the rest of the participants indicating that, on an initial validation, the classification model can be generic enough for any participant's accent.

Keywords: *lip reading, real time, computer vision, visual speech recognition, distributed system*

Contents

1 Introduction

Computer vision is today an effervescent field used in a lot of different domains of application. It is especially important to sense and perceive the environment around a robot to allow it to take decisions in every possible situations. By exploiting computer vision, social robots can achieve higher levels of autonomy and support humans in more cognitive tasks, such as localise themselves, recognise objects and detecting human intentions [1].

One of the main reasons for pursuing the implementation of algorithms to automate robots in human understanding tasks relies on the fact that robots capable of doing it can support teaching and learning processes. However, the difficulty in overcoming unexpected situations in the environment, such as noise for speech recognition, is constantly pointed out as one of the most challenging obstacle to be overcome by autonomous robots in educational context [2].

In this thesis, we are giving the first steps towards equipping social robots with state-of-the-art Visual Speech Recognition (VSR) for in-the-wild[1] conversation with acceptable processing time and accuracy, where by *acceptable* is implied that the delay in processing users' inputs and respond to them does not prejudice the human experience in interacting with such systems. The motivation is to deploy the resulting application in classrooms as on of the future steps. To simplify this complex problem, we will start by using a laptop instead of a robot and adults instead of children. To increase the accuracy of the Audio Speech Recognition (ASR), we will use lip reading since with this algorithms it has been demonstrated that, in extreme cases where the sound can absolutely not be used, it can still achieve a Word Error Rate (WER) of less than 40% [3].

We have one more constraint than those previously mentioned articles. As we are working on a machine having low computational power and aim real time applications, we want to minimize the resources taken while maximising the result accuracy. Knowing that all the articles of the systematic review [4] are using Deep Neural Networks (DNN) techniques and that non-DNN classifier were also tested on smaller datasets without achieving as good performance as DNN [5], we will use a DNN trained on the LRS3 dataset [6]. This dataset contains videos from the TED show saying over fifty thousand words. The challenge is to make it fast enough with a good precision.

[1] No predefined database or limitation for the predicted sentences.

2 Background

2.1 Autonomy in Social Robots and Childhood Education

We originally started this project investigating in the literature how computer vision impacts social robots for education and how to automate social robots can, in fact, enhance the learning experience and support teachers in their activities. We started by a systematic review, using the research criterion of Table 1 on IEEE and Scopus databases for articles published in 2021 and later. We found 17 articles but only 4 papers (shown in Table 2) were kept after applying the following exclusion criteria:

social	robot tutor	robot vision
		image processing
	robot teacher	teleoperated
		remote control
	educational robotics	Wizard of Oz

Table 1: Keywords used to find articles for the systematic review. The articles should contain one from each column.

- The robot is not used to teach.
- No experiment were done to validate.
- The experiment is not aiming children.
- The paper is not an article.
- No computer visions used.
- No social robots used.

Articles	**Use case**	**Robot**	**Autonomous**
[7]	detect and classify the waste material	Pepper	yes
[8]	Replicate the upper body of the teacher. Mindfulness training	Pepper	no
[9]	object detection	friendly dinosaur trash can	yes
[10]	help to teach autism spectrum disorder children	cozmo / mini-drone	both

Table 2: Systematic review articles.

For a social robot, it is crucial to perceive well its environment, so that it can have proper social behaviour without the need of a button or screen to make the interaction more natural to the human (e.g. answering or reacting to a human behaviour). Understanding the human gesture or speech, recognizing the human, having a reaction through a screen, a vocal or a movement are example of such desirable interactions. We can differentiate three types of interaction: robot to human (e.g. the robot speaks), human to robot (e.g. the robot understands what the human says) and both (e.g. a dialogue).

The outcome of this brief systematic review showed that often the used robots are not autonomous enough to perform without a human assistant next to it or remotely controlled as Wizard of Oz. This problem slow down the change from experiment and research to practicals active use of them in education. There are many reasons that make those robots not autonomous enough, as cited in [2]. One of them is that the audio speech recognition is not working well in noisy environment. By analysing works from lip reading area [4], we concluded that adding a visual classification on the lips may increase the speech prediction accuracy. Therefore, taking into consideration the gaps in how autonomy can better help social robots in classrooms, we decided to follow the lipreading direction. The following sections will describe how this work targets to tackle the investigated problems.

2.2 Lip Reading

Due to the large-scale data sets currently available [11, 12, 13] and the growing development of deep networks, it has been shown that DNN have the best accuracy for this type of classification problem [5]. Then this project is limited to this type of classifier.

Based on the systematic review of [4], we compare the two networks from [14] and [15]. Table 3 shows that they are similar but one is using one more BI-GRU layer to extract the temporal features, which makes it perform better in lip reading (82% vs 83.7% classification rate). Then comparing with two state-of-the-art works [16, 17], that both use Temporal Convolutional Networks (TCN) and give better accuracy. Furthermore, in [17] the authors tried to focus on the practical aspect of his network and manage to decrease its computational power requested which might be interesting for our real-time application. We can also compare it with [18] that claim to be able to run it in real time as an online application using long-short term memory (LSTM) networks, Convolutional Neural Networks (CNN) and Transformers [18]. The accuracy of this network is nevertheless not comparable with the previously mentioned network as it was trained on a the LRS2 dataset [19], which is a way more complex dataset based on video from the Oxford-BBC News. As a last comparison, we chose the state of the art Conformers network [3]. It has the best accuracy on LRS2 and LRS3 dataset and proves that doing VSR using this network will increase the accuracy of AVSR in noisy environment by showing how the noise is impacting the performances. For these reasons, we decide to continue our work using the Conformers network and will first focus on the VSR. Transformers are stacks of encoders layer and decoders layer, they achieve good performances at capturing content-based information. Conformers are Convolution-Augmented Transformer, where CNN exploiting local features are combined with transformers [20].

2.3 Face Tracking

An important step of lip reading is the face tracking and to do so, landmarks are used. They are points situated on predefined position of the face depending of the algorithm used. Those landmarks will be use to find the position of the mouth in

Net	Date	Visual feature extraction	Temporal feature extraction	Dataset	Accuracy
[14]	22.02.18	resnet-34	2xbigru (1024)	lrw	82% CR
[15]	15.03.20	resnet and others	3xbigru (1024)	lrw (-lrw-1000)	83.7% CR
[16]	23.01.20	resnet-18	MS-TCN	lrw (-lrw-1000)	85.3% CR
[17]	02.06.21	shuffleNetV2	DS-TCN	lrw	88.5% CR
[18]	15.06.18	resnet	lstm/CN/transformer	lrw/LRS2	lstm : 35.3% CER
[3]	11.06.21	resnet-18	conformer	LRW + LRS3	43.3% WER

Table 3: Comparison of different models (CR is the classification rate)

.

the image to be able to crop around it and these cropped images will be used as input to the VSR network.

We considered many different possibilities, such as MediaPipe[21], Dlib[22] or the Haar Cascade Classifier of OpenCV[23]. However, MediaPipe was chosen as the most affordable solution since is faster than Dlib and less dependant of the luminosity than the OpenCV Cascade Classifier. Furthermore, we tested whether the precision of MediaPipe has acceptable trade-off between prediction time and accuracy. We also considered some different options to build our own system, for instance to use optical flow, to delete the background of the image or to make the mouth rotational invariant by using the face landmarks to rotate the image so that the mouth is horizontal and that the eyes, ears and nose are above it. The other ideas were abandoned for being not suitable for the scope of the application after a fast initial validation. Even if the optical flow could be an interesting non explored solution for lip reading, we realised that some good VSR algorithm already exist and decided to use one of them. Deleting the background will not help at all as almost all good existing model crop the image around the mouth before sending it to the network.

The state-of-art VSR [3] that we are considering in our implementation uses RetinaFace [24] as a first preprocessing step, followed by a Face Alignment Network (FAN) [25] to obtain 68 reliable landmarks of the face over the video. We first adapted this system to be able to run on CPU. But then realised that we need something faster. This is why we decided to use the MediaPipe Face Mesh detection. This algorithm is, however, giving 468 landmarks. To make sure that it is well integrated in the algorithm, we mapped the closest point of MediaPipe to the each Retina Fan landmarks (see Figure 1).

Therefore, the code from [3] is taking care of interpolating the frames where no faces are detected, skipping cases where we will not be able to lip read or if the video is too short, and then, the algorithm smoothies the landmarks on a window range and finally crop the mouth region.

2.4 Synthesis

We decided to build a distributed VSR system taking the code that used Conformers as VSR algorithm from the proposal of [3]. However, its prediction time is too long in computers that have no GPU, such as computers used by robots or robotic embedded systems themselves. This fact may prejudice the human experience when trying to communicate with autonomous systems. One of the main contributions of this work is the adaptation to do so by implementing the pipeline in a distributed system, as

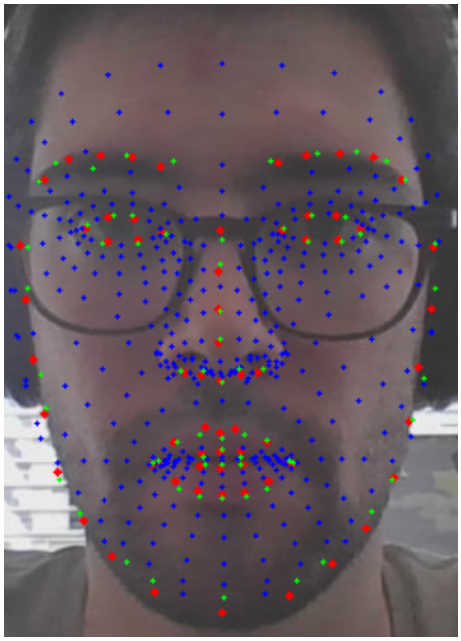

Figure 1: The red points are Retina Fan's landmarks. The blue and green points are the 468 landmarks of MediaPipe. Each Retina FAN point (red) has a MediaPipe point (green) associated, which is its closest point.

explained in Section 4.

3 Experiment Design

3.1 Research Questions

In this study we want to compare different implementations and methods for a distributed architecture aiming to achieve faster processing time and higher classification performance in state-of-art algorithms for VSR. To this aim we have the following research questions:

RQ1: Pipeline distribution
How does the division of the pipeline for VSR in distributed processing nodes impact the time to process and Word Error Rate (WER) measurement?

RQ2: Boosting preprocessing time
How does the use of different preprocessing algorithms, such as MediaPipe, affect the VSR performances?

RQ3: Users Nationalities
Is the lip reading algorithm accuracy sensitive to the users' mother-tongues?

3.2 Validation Design

To investigate our research questions used 25 sentences taken from a list of English idioms [26] (shown in appendix A) covering all the visemes defined by Amazon Polly [27]. The distribution of the different visemes can be observed in Table 3.2. To those 25 sentences were added five more that each participants had to think and write themselves. We invited people from diverse nationalities to speak these sentences. A total of 570 videos of 19 different people speaking the sentences were collected and used for the validation.

We compare the performance measures (explained in Section 3.6) of the videos of the subjects reading those sentences according to the conditions shown in Table 5.

The hardware configuration of the computers used in the experiments, laptop and server (desktop), are the following: The CPU of the laptop is an Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz x 4, the CPU and GPU on the server are, respectively, an Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz x 8 and an NVIDIA GeForce GTX 1080 Ti.

3.3 Participants

The participants were 19 adults from EPFL. Seven of them were English natives from Britain or United States while 12 were coming from all around the world but English speaking countries. The experiment was conducted during a week where each of them participated individually during the time slot they chose. They were recruited as volunteers through direct discussions or mails where they were asked if they wanted to help us to evaluate the performances of our VSR system aiming to work in real time.

Visemes	Phonemes	Exemples	Distribution
@	@,@U	**a**ren**a**, g**oa**t	10,4
a	{, aI, aU, A:	tr**a**p, pr**i**ce, m**ou**th, f**a**ther	7,3
e	eI	f**a**ce	2,7
E	3:, E, E@, V	n**ur**se, dr**e**ss, squ**are**, str**u**t	2,3
f	f, v	**f**ive, **v**est	3,5
i	j, i, I, I@	**y**es, fl**ee**ce, k**i**t, n**ear**	10,8
k	g, h, k, N	**g**ame, **h**ouse, **c**at, thi**ng**	8,7
O	O:, OI, Q	th**ou**ght, ch**oi**ce, l**o**t	0,6
p	b, m, m=, p	**b**ed, **m**ouse, anth**em**, **p**in	5,2
r	r/	**r**ed	5,4
s	z, s	**z**ero, **s**eem	7,9
S	S, tS, dZ, Z	**sh**ip, **ch**art, **j**ump, **v**ision	1,9
t	d, l, l=, n, n=, t	**d**ig, **l**ay, batt**le**, **n**ap, butto**n**, **t**ask	22,0
T	D, T	**th**en, **th**in	4,2
u	w, u:, U, U@	**w**est, g**oo**se, f**oo**t, c**u**re	7,1

Table 4: Relations between visemes and X-SAMPA phonemes. The examples highlights respectively the phonemes represented. The distribution column represents the percentage that each phoneme has in the sentences prepared for the experiment out of a total of 519 visemes.

#	Landmark extraction	Processing type
1	Retinafan	CPU
2	Retinafan	GPU
3	Mediapipe	CPU
4	Mediapipe	GPU

Table 5: Conditions to be validated.

3.4 Data Collection

To record the videos, the participants were placed in front of the laptop in a silent environment and asked to read the sentences shown on the screen. Even if they know it was an algorithm trying to lipread, they were asked to speak as normally as possible.

We asked the participants about their mother tongue to validate whether being a English native speaker would interfere on the classification results.

During the reading, the participants had to press the "space" button to start the recording of their image and sound, read the sentence shown and press again the "space" button to stop the recording. They could then start again with the next sentence. If the reading was too much failed, having stuttering or missing a word, the participant was asked to redo the video. At the end of the experiment, thirty videos per participants were collected.

3.5 Hypotheses

We expect to decrease the processing time of the VSR by using a distributed architecture where the classification node runs over GPUs so that it would be possible to use it in real time, meaning that the user should only wait a few seconds before the robot answer and will not impact the fluency of the Human-Robot Interaction (HRI). We also expect to not significantly decrease the accuracy of the initial system because a good HRI requires that the robot understands well the user.

Using the GPU of a computer will decrease the inference time and actually it became a common habit to not make the heavy computational algorithm on robot but to do it on a server where more computational power are available. This is why, for RQ1, we expect the computation to be faster by dividing the VSR pipeline in distributed nodes. Furthermore there is no reason that it impacts the accuracy of the algorithm, so we do not expect any changes on the WER. Per consequence, our first hypothesis is:
H1 - Distributing the pipeline will decrease the time of feed forward and will not impact the accuracy.
The duration of the face landmarks detection using Retina FAN done in this VSR algorithm is actually taking more time than the inference part. The precision and reliability of this algorithm might be expected for some purposes, but in our circumstances, we would like to decrease more the processing time. For RQ2, by using MediaPipe instead of Retina FAN, we expect a big upgrade in processing time but a worth precision. Whatever, we do not expect this decrease of precision to impact a lot the system as it is only used to crop the mouth region doing a mean of the landmarks, before using the cropped image as input of the neural network. Our second hypothesis is then :
H2 - Using MediaPipe instead of Retina FAN will also decrease the processing time required and might present slightly lower accuracy.
As the VSR networks was trained on LRS3 BBC dataset (videos of only British English speakers), for RQ3, we expect that it will achieve a better precision for English native speaker than for the others. So our third hypothesis is:
H3 - English native speaker participants will present a better accuracy in the lip reading.

3.6 Evaluation Metrics

Playing the videos recorded during the experiment with the different system configurations explained in Section 3.4, we will investigate the research questions and validate whether our hypotheses were correct. For this purpose, we will use the metrics described in the following sections.

3.6.1 Speed

To measure the speed of the algorithm, we will simply measure the time between the different step of the system. The time steps measured are the duration of the video (recording time), the time it takes to transfer the audio and visual data to the

server (transferring time), the time it takes to build the video based on the audio and visual data received (video building time) and finally the time it takes to get the landmarks, crop the image and infer the results (inference time).

3.6.2 Accuracy

To measure the accuracy of the system, we use the Word Error Rate (WER) and the Character Error Rate (CER). The lowest those measures are, the better it is. These measured are obtain according to Equations (1, 2) :

$$WER = \frac{Substitutions + Insertions + Deletions}{Number\ of\ Words\ Spoken} \tag{1}$$

and

$$CER = \frac{Substitutions + Insertions + Deletions}{Number\ of\ Character\ Spoken} \tag{2}$$

Where a substitution occurs when a word/character gets replaced by another one, an insertion when it is added and a deletion when it is not in in the result.

4 System Architecture

Knowing that the aim is to help social robots to understand better human speaking in noisy places, the system needs to be compatible to robot software environment. We also need to keep in mind that robots and laptops hardware normally do not have a lot of computational power and doing precise VSR as described in Section 2.2 will ask some computational power. So we are investigating whether distributed computing (processing heavy data in a remote and more powerful hardware) can minimize this issue. To do so all the system is made using the Robot Operating System (ROS). ROS allows to create nodes which have one or many tasks. The nodes can publish and subscribe to topics to respectively write or read their content. The subscribers define a callback function that is called every time a node publish in that topic. It allows the nodes to communicate efficiently and to have multiple processes running at the same time. They also have common parameters called rosparam. Those parameters can be written in a configuration file and loaded at launch to give access to every nodes.

The system contains four nodes/modules and an additional one for evaluation. Those nodes are listed below and explained in details in their corresponding Subsection 4.1 to 4.5. The nodes, where they should run and theirs corresponding functions are briefly described next and detailed in the next subsections.

- ***State Manager*** [Running on the server]: Ensures that the actual state is well completed before deciding and setting the correct following state. It keeps the process timeline correct at all time and avoid being in non-predicted states.
- ***Visual module*** [Running on the laptop]: responsible of taking images and to make sure that their transfer to the Video Module was correctly done.
- ***Audio module*** [Running on the laptop]: responsible of capturing sounds and to make sure that their transfer to the Video Module was correctly done.
- ***Video Module*** [Running on the server]: first responsible of receiving the images and sounds sent by the Visual and Audio modules to build a video out of them and then of using the video to get the prediction of the speech of it.
- ***Evaluator*** [Running on the laptop]: feature added so that, instead of using the camera and microphone input, the system will use an mp4 video input. It will also take care of saving the processing time and results obtained.

The implementation of the architecture this way was important to guarantee efficient communication between the nodes scattered in the different computers. As shown in Figure 2, the nodes are distributed between the laptop and the server. This system was designed to be used on a robot and to do AVSR even if in the scope of this report we will use a laptop and a VSR system. The Vision and Audio Module are running independently on the laptop and could be used on different one if needed. Nevertheless, when running the evaluator, it needs to be on the same computer than the audio and visual module, which constraints their independence. The Video Module run on the server as it contains the most computational algorithm and finally, the state manager could run anywhere but we suggest to also run it on

the server. Even if it is not doing any computational operation we don't want to add any computation to the laptop.

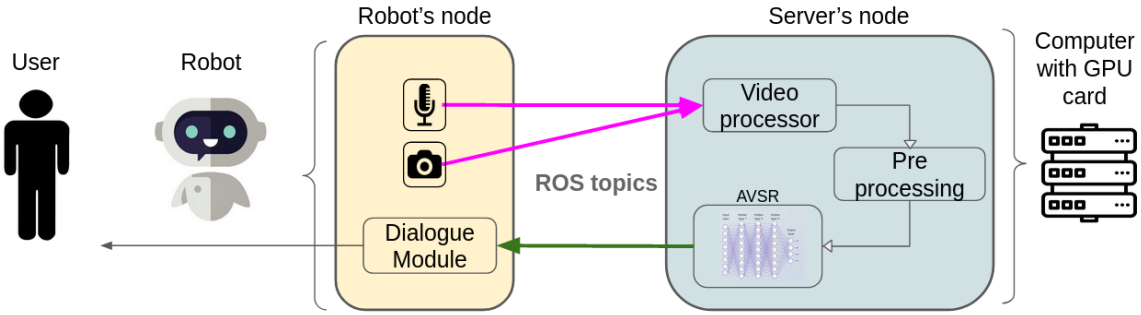

Figure 2: Overview of all the elements interaction in the system designed to use AVSR on a robot/laptop. In the yellow box are the nodes running on the robot (or the laptop connected to the robot). They are the Visual and Audio modules, used to capture the data then send to a server which will be a more powerful system. It also contains the dialogue module that will receive the results of the AVSR to use it for its task. The dialogue module is not part of this project but future steps. On the server side, the blue box has the Video Module node. It will process the data received by the robot to give the the format needed for the following steps. Those steps are the preprocessing which will extract the landmarks of the mouth and get the mouth region in the image and then send it to the AVSR algorithm to finally be classified.

Details of ROS nodes implementation are shown in the Appendix through rqt_graphs[2] images.

4.1 State Manager

This module controls the data flow and decision-making of the state transition in the architecture. There are three different timelines/modes, the **normal**, the **experiment** and the **evaluation**. Their differences can be seen in Figure 3) and is explained in the next paragraphs. During the normal timeline, we want to understand the user speech using live streaming of the camera and the microphone in real time. Starting from Idle state, we can jump to the Recording state as soon as everything is ready, which can take a few seconds because of the loading of the neural networks used for face tracking and lip reading. We can then stop the recording so that the Transferring state start. The latter ensures that all the data are well arrived before starting the Video Building state. Once that the Video is built, the system start its last task which is the Video Processor state.
While in experiment timeline, it will stay the same except that it will skip the video process. The goal is to only record the live streaming to collect data that can be used later.
And finally, the evaluation timeline is used to evaluate the performances on some videos in all the system. It is important to make it as if the videos are the input

[2] http://wiki.ros.org/rqt_graph

of the Visual and Audio modules. The videos can for example be the ones collected during experiment. During evaluation, the Idle and Recording states will be replaced by the Evaluation_idle and Evaluation_recording states. This gives the information to the Visual and Audio modules that they should use a video file as input instead of the live stream.

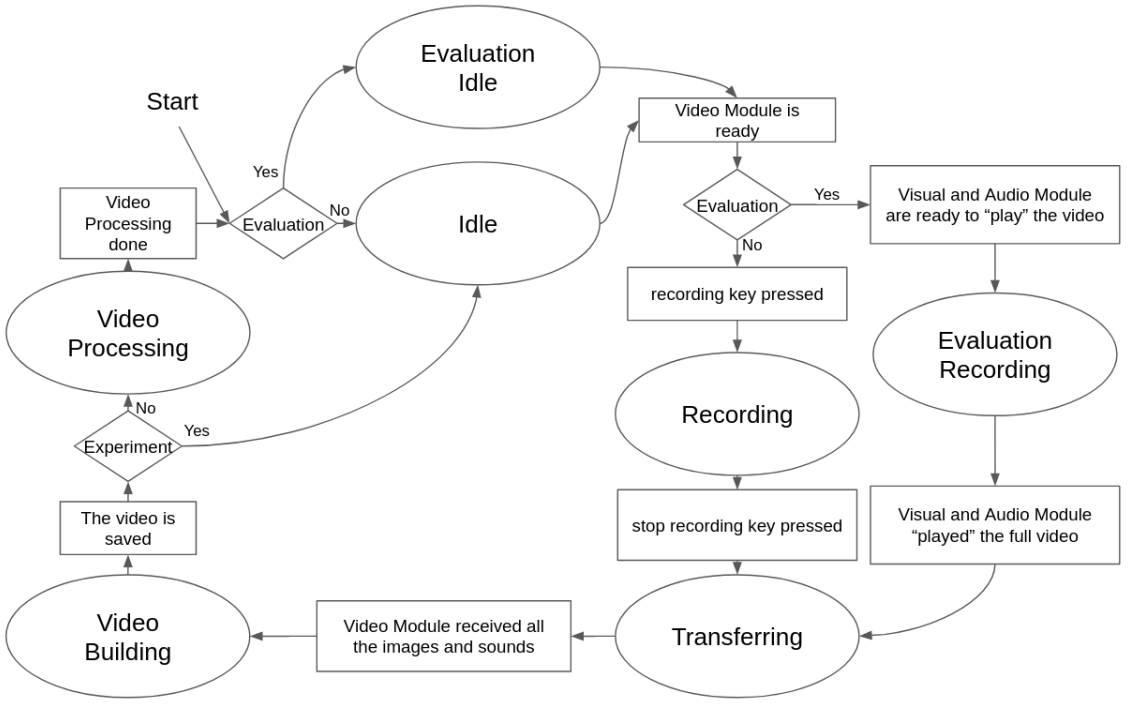

Figure 3: Shema of the flow between the states. Circles are states, rectangle are conditions to switch states and diamonds are conditions to determine which is the following state. The start is at the top left, it is not a state.

To sum up, the state manager receives informations that help it to determine if it has to change the state of the system or not. (For more details, see Figure 5)

The Recording and Transferring state will be explained in section 4.2 and 4.3. The Video Building and Video Processor will be explained in the section 4.4.

4.2 Visual Module

The Visual module is taking care of the visual input of the system. It is also the part that will get user inputs allowing him to start and stop the Recording. When the state becomes Recording, it will start sending the image to the topic matching the configuration request. The image could be a compressed image or a raw image eventually combined with face landmarks. The compressed image is probably always preferred compared to the raw image as it is transferred faster. Doing the landmarks detection at this step while waiting to send the next image avoid to do it during the video process, it might make win a small amount of computation time but in return add some computations on the laptop. During all the Recording state it will count

the number of images sent while the Video module sends back the number of frame received so that when the Transferring state start, the Visual can control that all images were correctly sent before confirming to the State Manager that all images were transferred.

In evaluation mode, the user is not allowed to interact with the system. It will load the video sent by the evaluator and then automatically requests to start and stop the Recording.

4.3 Audio Module

At the start of the Recording state, it starts sending chunks of audio data and counting how much chunks were sent. When switching to Transferring, it verifies that the number of chunk received by the Video Module is the same as the number of chunk sent to transmit to the state manager when the transfer is complete. The Audio can only make a recording request, when it is in evaluation mode. This means that it correctly received which video it has to play the sounds from and that it is ready to play it, while in normal mode, it is just waiting that the state change to start the recording.

4.4 Video Module

This node is divided in two parts. The first one explained in Section 4.4.1 builds the video and is followed by the second part developed in section 4.4.2 which will process it. As it needs to load the models, it takes a bit of time to initialise it, so the State Manager wait to receive a confirmation that this node is ready to receive some data before going out of Idle.

4.4.1 Video Builder

Active during the Recording and Transferring, it receives the data sent by the Visual and Audio modules and send back the quantity of data received. Once those step validated, the State Manager switches in Video Building, so this module now builds the video and then tell the State Manager when this is done so that the state manager knows that the video process can start.

4.4.2 Video Processor

We now have a video and only need to use the code related to [3] to process it. Using the images for the speech recognition, it first searches the face landmarks in every images of the video (see Section 2.3). Then it uses them to crop the mouth region in every frames and finally send those mouth region in the network (see Section 2.2). The network returns the wanted prediction of the speech said in the video.

4.5 Evaluator

The goal of this part is simply to make a smooth process to evaluate how the system performs on videos. At initialisation it gets the path of all the videos that it should

evaluate and will then make the system evaluate them one per state cycles. During the state cycles it takes the timing and accuracy measures wanted. At every end of cycles, it saves the results in a csv file. For the purpose of this report, the saved elements are the following:

- Participant: Keeps track if it is an English native speaker.
- Preprocessing: Keeps track of the face landmarks detection algorithm used (see Section 2.3 for more details on them).
- Recording [s]: The time it took to record the video. It is close to the duration of the video.
- Transferring [s]: The time it took to end the transfer after that the recording ended.
- Video Building [s]: The time it took to build the mp4 video using the data received.
- Video Processing [s]: The time it takes for the video building, face tracking and inference.
- WER: World Error Rate
- Nb word: Number of worlds in the ground truth sentence. Allows to compute the WER of all sentences together.
- CER: Character Error Rate
- Nb character: Number of characters in the ground truth sentence. Allows to compute the CER of all sentences together.
- Ground truth: The sentence read in the video.
- Prediction: The prediction that the algorithm made on the speech of the video.

5 Results

The results can be separated by metrics regarding the system performance of speed and accuracy. The results of speed will allow us to determine how much we were able to speed up the process while the results of accuracy will show us the impact of those modifications and of the accent of the participants.

5.1 Speed

5.1.1 Face's landmarks Detection

To compare the results of speed in different nodes of processing the classification algorithm and different preprocessing methods, we ran the whole loop of classifying the sentences for all the videos of one participant (participant 3) completely in a laptop and using the distributed architecture to compare the different outcomes between condition 1 and 2 and condition 3 and 4 of Table 5.

By using MediaPipe instead of Retina FAN, it can be seen in Table 6 that when only using the laptop, the times of the different steps stay the same except for the video processing time. This behaviour is corresponding with the initial expectations. The video processing time is then seven times smaller for MediaPipe, confirming our second hypothesis that MediaPipe is decreasing the processing time.

Preprocessing	**Recording [s]**	**Transferring [s]**	**Video Building [s]**	**Video Processing [s]**	**Total**
MediaPipe	3.13 ± 0.53	0.03 ± 0.01	1.32 ± 0.37	20.650 ± 8.467	25.14 ± 9.37
Retina FAN	3.12 ± 0.52	0.03 ± 0.01	1.33 ± 0.18	142.391 ± 27.858	146.87 ± 28.57

Table 6: Results achieved using the videos of the participant number 3 on the laptop. (Average ± standard deviation)

5.1.2 Distributed System

By comparing Table 6 and 7, we can observe that the transferring time has a slightly increase while the video building and processing time both decreased. As the system now have to transfer the data through the network to have more computational power during the building and processing of the video, this is the result expected by H1. In total, it was initially using Retina FAN on the laptop having an average time of about 147 seconds that we were able to decrease by a factor 24.5 to finally achieve a process taking 6 seconds, but only 2.87 seconds from the end of the recording to the moment were the prediction is done. By running an unpaired t-test on the video processing time of Table 12 and 13, we achieved a two-tailed p-value of less than 0.0001. This difference is considered to be extremely statistically significant.

Preprocessing	**Recording [s]**	**Transferring [s]**	**Video Building [s]**	**Video Processing [s]**	**Total**
MediaPipe	3.12 ± 0.52	0.05 ± 0.02	0.57 ± 0.05	2.25 ± 0.49	5.99 ± 1.08
Retina FAN	3.13 ± 0.52	0.04 ± 0.02	0.54 ± 0.04	5.49 ± 1.02	9.20 ± 1.60

Table 7: Results achieved using the videos of the participant number 3 on the server. (Average ± standard deviation)

5.2 Accuracy

By looking at Table 8, we can first remark that the average WER is a bit lower using the Retina FAN than MediaPipe (0.650±0.213 vs 0.684±0.259) but looking at the standard deviation, this is not significant. Nevertheless, by looking at the participant number 5, we can see that there was a problem when using MediaPipe. The face was not detected or detected at the wrong place in all the video while Retina FAN detected him. We only experienced this problem with the datasets of this participant. This under performance can eventually be explained by the luminosity of the video. The background of those videos were more saturated than in the others. This is the kind of problems that we could have expected in H2 and shows us the limitation of MediaPipe compared to Retina FAN. By running an unpaired t-test on the WER of Table 12 and 13, we achieved a two-tailed p-value of 0.5878. This difference is considered to not be statistically significant.

Now looking at the green versus white regions of the same tables, knowing that the green part are for the English native speaker, it can be observed that the English native speaker achieve better WER and CER than the others. As the NN was trained on English native speaker, this was a result expected in H3. Nevertheless, some English native speaker still doesn't perform as well as they should. For the participant number height, the background was more complex than in the other videos, while for number 6 and 7 the background was darker, so we can one possible explanation is that the face landmarks could have been wrong in some videos. Another explanation could come from the fact that the videos of number 7 and 8 are shorter than the others which might imply that they were speaking faster and might impact the accuracy of the NN.

The accuracy may also be perturbed by the fact that all the participants knew that they were recorded for an algorithm doing lips reading, so they may have unconsciously increase their articulation and we could expect that it shifts a bit the results in a positive or negative way.

In Table 9, we see that the results are lightly better than in Table 8. This is known that GPU and CPU produce slight differences in results due to rounding decimal places. Those differences are not relevant in our case.

Participant	WER	CER
1	0.275	0.171
2	0.311	0.186
3	0.337	0.216
8	0.353	0.219
9	0.495	0.318
5	0.650	0.469
11	0.655	0.450
10	0.673	0.457
12	0.682	0.482
6	0.696	0.493
13	0.766	0.531
7	0.774	0.544
14	0.811	0.613
15	0.812	0.594
16	0.84	0.620
17	0.877	0.676
18	0.911	0.692
19	1.054	0.846
4	1.315	1.017
AVERAGE	0.699 ± 0.265	0.505 ± 0.223

Participant	WER	CER
1	0.286	0.195
2	0.321	0.197
3	0.352	0.220
8	0.363	0.234
4	0.462	0.291
9	0.542	0.367
10	0.617	0.394
5	0.635	0.449
11	0.706	0.522
12	0.719	0.520
6	0.722	0.515
13	0.785	0.585
14	0.787	0.558
7	0.795	0.533
15	0.796	0.593
16	0.814	0.604
17	0.862	0.651
18	0.896	0.678
19	1.016	0.865
AVERAGE	0.657 ± 0.213	0.472± 0.185

Table 8: On the left side is the accuracy obtained using MediaPipe on the server. On the right side is the accuracy obtained using Retina FAN on the server. The green rows are participants that are English native speaker.

Preprocessing	Total WER	Total CER
MediaPipe	0.301	0.197
Retina FAN	0.281	0.188

Table 9: Results achieved by classifying the videos of the participant number 3 only on the laptop.

6 Discussion

6.1 Technical Challenges

The first technical challenge we encountered was to transfer efficiently the images and sounds from the laptop (less powerful computer) to the server. Instead of establishing a VPN, we used the Husarnet Client app [28]. It allows us to have topics communicating in both directions between the computers.

The second technical challenge encounter was a misunderstanding with the algorithm that we chose to do AVSR. We thought until a few weeks before the end of the project that we could use it for that purpose as they were results using it in the paper and part of its implementation in the code. But in fact, it was no more usable for AVSR, but only for VSR. So we did not manage to use both images and sounds information at the same time and needed to switch to another directions of the project which was only relying on the images. This fact had a bad impact on the implementation timeline.

6.2 Experiment Challenges

The only problem encountered is that the participants were not speaking at the same voice level and at the same distance of the computer. It has no influence on our work as we are not using the sound. However, that led to some videos where we barely hear the participant and some others where the sound is saturated. It shows that having the images can help even in circumstances that are not only linked to external noise.

6.3 Future Directions

Another aspect of the project that is not validated but is implemented is to compute the landmarks on the laptop and send them and the video to the server. As MediaPipe is extremely fast, it will not slow down the 30 fps that we wanted to have. So it would be interesting to check if it helps going even faster. In an extreme scenario, we can even try to go further and directly find the ROI on the laptop to only send the mouth region image to the server and see how much faster it is. But coming back to our willing of integrating this on a robot, all this would have the cost to have less computational power for other purposes.

We also wanted to test the set up using an AVSR model. But there was a misunderstanding of the algorithm we use and it was not possible to do it during the project period. So a future interesting direction would be to test it with an AVSR model or to find a way to combine the results of the report with a good ASR model, to check if the speed is still fast enough and how much useful the VSR is when adding different noise level to the audio.

We now intent to test and validate our system to understand children speaking. But as the results with the different accent shown, we expect that we should find a way to retrain the networks using children speaking dataset to stay accurate enough.

We started to go toward more directions to speed up the process, which is to decrease the frame per seconds (fps) of the videos. Testing it with 44 videos recorded on two non-English native speaker at 15 fps, we obtained the results of Table 10. Comparing with the Tables 7, it is possible to see that the Video Processing time is smaller while the WER and CER are visually worse but close to the average of the ones found in Table 8. We don't expect to have WER and CER equal to the average of Table 8 as this average is computed with participants that are English native speakers. It could be, therefore, a nice way to further improve the processing time of the algorithm to be validated on a bigger dataset.

Preprocessing [s]	**Recording [s]**	**Transferring [s]**	**Video Building [s]**	**Video Processing [s]**	**WER**	**CER**
MediaPipe	2.48 ± 1.13	0.03 ± 0.01	0.54 ± 0.12	2.05 ± 0.66	0.73 ± 0.38	0.53 ± 0.28
Retina FAN	2.55 ± 1.21	0.03 ± 0.01	0.54 ± 0.13	5.01 ± 2.23	0.74 ± 0.39	0.54 ± 0.30

Table 10: Results achieved by running the 15 fps videos of non-English native speaker on the server. (Average ± standard deviation)

7 Conclusion

In this thesis, we explored the first steps in giving robots better speech recognition tools in noisy environment, by proposing a distributed architecture system for VSR that can be combined with ASR to increase their accuracy. To this goal, we decided to use VSR implemented in a distributed system using ROS. Results show an increase of the speed of the system by 15 times when clarifying sentences in the server compared to classifying the sentences in the laptop node. We also changed the face landmarks detection used from Retina FAN to MediaPipe to further increase the speed of the system. This modification sped up the system by a factor 2 and did not significantly decreased the WER accuracy when using Mediapipe. Nevertheless, it decreased the reliability of detecting faces and when it is not detecting them well it makes the lip reading impossible or giving bad results. We finally realised how much the mother-tongue of the participants had an impact on the accuracy of the system. In general, English native speaker participants achieved 12% smaller WER than non-native English speaker ones. So working with children dataset to first test the system and if it is not working retraining the network using those dataset might be an interesting future for the project.

The code developed during this thesis can be find on the GitHub repository of the CHILI lab [29].

References

[1] Aphrodite Sophokleous et al. "Computer vision meets educational robotics". In: *Electronics* 10.6 (2021), p. 730.

[2] Tony Belpaeme et al. "Social robots for education: A review". In: *Science robotics* 3.21 (2018), eaat5954.

[3] Pingchuan Ma, Stavros Petridis, and Maja Pantic. "End-to-end audio-visual speech recognition with conformers". In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 7613–7617.

[4] Marzieh Oghbaie et al. "Advances and Challenges in Deep Lip Reading". In: *arXiv preprint arXiv:2110.07879* (2021).

[5] Adriana Fernandez-Lopez and Federico M Sukno. "Survey on automatic lip-reading in the era of deep learning". In: *Image and Vision Computing* 78 (2018), pp. 53–72.

[6] T. Afouras, J. S. Chung, and A. Zisserman. "LRS3-TED: a large-scale dataset for visual speech recognition". In: *arXiv preprint arXiv:1809.00496*. 2018.

[7] Giovanna Castellano et al. "PeppeRecycle: improving children's attitude toward recycling by playing with a social robot". In: *International Journal of Social Robotics* 13.1 (2021), pp. 97–111.

[8] Indu P Bodala, Nikhil Churamani, and Hatice Gunes. "Teleoperated robot coaching for mindfulness training: A longitudinal study". In: *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*. IEEE. 2021, pp. 939–944.

[9] Saifuddin Mahmud et al. "An Intelligent Trash Can Robt for Early Childhood Green Education". In: *2021 16th International Conference on Computer Science & Education (ICCSE)*. 2021, pp. 870–874. DOI: 10.1109/ICCSE51940.2021.9569360.

[10] Nabanita Paul et al. "Can Non-Humanoid Social Robots Reduce Workload of Special Educators : An Online and In-Premises Field Study". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 2021, pp. 11386–11392. DOI: 10.1109/ICRA48506.2021.9561633.

[11] J. S. Chung and A. Zisserman. "Lip Reading in the Wild". In: *Asian Conference on Computer Vision*. 2016.

[12] J. S. Chung et al. "Lip Reading Sentences in the Wild". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

[13] J. S. Chung and A. Zisserman. "Lip Reading in Profile". In: *British Machine Vision Conference*. 2017.

[14] Stavros Petridis et al. "End-to-end audiovisual speech recognition". In: *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2018, pp. 6548–6552.

[15] Dalu Feng et al. "Learn an effective lip reading model without pains". In: *arXiv preprint arXiv:2011.07557* (2020).

[16] Brais Martinez et al. "Lipreading using temporal convolutional networks". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 6319–6323.

[17] Pingchuan Ma et al. "Towards practical lipreading with distilled and efficient models". In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 7608–7612.

[18] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. "Deep lip reading: a comparison of models and an online application". In: *arXiv preprint arXiv:1806.06053* (2018).

[19] T. Afouras et al. "Deep Audio-Visual Speech Recognition". In: *arXiv:1809.02108*. 2018.

[20] Anmol Gulati et al. "Conformer: Convolution-augmented transformer for speech recognition". In: *arXiv preprint arXiv:2005.08100* (2020).

[21] Google. *MediaPipe*. 2020. URL: https://google.github.io/mediapipe/ (visited on 06/21/2022).

[22] Google. *Dlib*. 2022. URL: http://dlib.net/ (visited on 06/22/2022).

[23] OpenCV. *Face Detection using Haar Cascades*. 2022. URL: https://docs.opencv.org/3.4/d2/d99/tutorial_js_face_detection.html (visited on 06/22/2022).

[24] Sefik Ilkin Serengil and Alper Ozpinar. "HyperExtended LightFace: A Facial Attribute Analysis Framework". In: *2021 International Conference on Engineering and Emerging Technologies (ICEET)*. IEEE. 2021, pp. 1–4. DOI: 10.1109/ICEET53442.2021.9659697. URL: https://doi.org/10.1109/ICEET53442.2021.9659697.

[25] Adrian Bulat and Georgios Tzimiropoulos. "How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks)". In: *International Conference on Computer Vision*. 2017.

[26] Education First. *English idioms*. 2022. URL: https://www.ef.com/wwen/english-resources/english-idioms/ (visited on 06/23/2022).

[27] Amazon Web Services. *English, British (en-GB)*. 2022. URL: https://docs.aws.amazon.com/polly/latest/dg/ph-table-english-uk.html (visited on 06/23/2022).

[28] Husarion. *Running ROS on multiple machines*. 2022. URL: https://husarion.com/tutorials/ros-tutorials/5-running-ros-on-multiple-machines/ (visited on 06/23/2022).

[29] Daniel Carnieto Tozadore David Roch and Pierre Dillenbourg. *CHILIpReading*. 2022. URL: https://github.com/CHILIpReading (visited on 06/23/2022).

[30] Xiangyu Zhang et al. "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.

A Experiment Sentences

1. A bird in the hand is worth two in the bush
2. Actions speak louder than words
3. A picture is worth a thousand words
4. Better late than never
5. Dont count your chickens before they hatch
6. Do unto others as you would have them do unto you
7. Every cloud has a silver lining
8. Good things come to those who wait
9. He has bigger fish to fry
10. Hes a chip off the old block
11. Hit the nail on the head
12. Ignorance is bliss
13. It aint over till the fat lady sings
14. Its a piece of cake
15. Its raining cats and dogs
16. Live and learn
17. Look before you leap
18. No pain no gain
19. Slow and steady wins the race
20. The devil is in the details
21. The robot cant make an omelet without breaking some eggs
22. Time flies when youre having fun
23. Well cross that bridge when we come to it
24. You cant judge a book by its cover
25. Your guess is as good as mine

B Additional results

B.1 Retina FAN vs MediaPipe on laptop

Preprocessing	Recording [s]	Transferring [s]	Video Building [s]	Video Processing [s]	WER	CER	Groundtruth	Prediction
mediapipe	3.47	0.07	2.33	23.50	0.250	0.222	HOW MUCH IS THAT DOGGY IN THE WINDOW	HOW MUCH IS THAT TALKING IN THE WINDOWS
mediapipe	2.47	0.03	0.97	11.34	0.000	0.000	LOOK BEFORE YOU LEAP	LOOK BEFORE YOU LEAP
mediapipe	2.47	0.03	0.97	13.63	0.500	0.462	MAKE LIKE A TREE AND LEAVE	BACK ON A JOURNEY AND LEAVE
mediapipe	3.07	0.03	1.07	16.90	0.000	0.000	A PENNY SAVED IS A PENNY EARNED	A PENNY SAVED IS A PENNY EARNED
mediapipe	3.13	0.03	1.07	12.74	0.000	0.000	EVERY CLOUD HAS A SILVER LINING	EVERY CLOUD HAS A SILVER LINING
mediapipe	3.60	0.03	1.23	17.00	0.714	0.393	HES A CHIP OFF THE OLD BLOCK	IS THE SHIP OF THE NORTH BLOCK
mediapipe	3.30	0.03	1.07	27.10	0.000	0.000	DONT COUNT YOUR CHICKENS BEFORE THEY HATCH	DONT COUNT YOUR CHICKENS BEFORE THEY HATCH
mediapipe	3.63	0.03	1.27	23.30	0.875	0.556	IT AINT OVER TILL THE FAT LADY SINGS	AND HEAD OVER TO THE FACT THAT HE SEES
mediapipe	2.97	0.03	1.20	18.13	0.000	0.000	THE DEVIL IS IN THE DETAILS	THE DEVIL IS IN THE DETAILS
mediapipe	2.80	0.03	1.17	21.87	0.667	0.607	THE EARLY BIRD GETS THE WORM	THE HALL APPEARED AGAINST THE WALL
mediapipe	2.87	0.03	1.10	13.04	0.200	0.158	ITS A PIECE OF CAKE	ITS A PIECE OF AIR
mediapipe	3.17	0.03	1.23	16.84	0.200	0.120	ITS RAINING CATS AND DOGS	ITS READING CATS AND DOGS
mediapipe	3.33	0.03	1.17	14.90	0.000	0.000	SLOW AND STEADY WINS THE RACE	SLOW AND STEADY WINS THE RACE
mediapipe	2.43	0.03	1.00	10.03	1.000	0.600	NO PAIN NO GAIN	YOU PAN YOUR HAND
mediapipe	2.27	0.03	0.93	9.63	0.000	0.000	LIVE AND LEARN	LIVE AND LEARN
mediapipe	2.27	0.03	1.00	11.40	0.750	0.364	BETTER LATE THAN NEVER	BIGGER LIGHT THAN EVER
mediapipe	4.17	0.03	1.47	40.10	0.182	0.102	DO UNTO OTHERS AS YOU WOULD HAVE THEM DO UNTO YOU	TO OTHERS AS YOU WOULD HAVE THEM DO UNTO YOU
mediapipe	2.87	0.03	1.20	13.70	1.333	0.833	IGNORANCE IS BLISS	AND CREATES THIS PLACE
mediapipe	3.67	0.03	1.50	25.63	0.000	0.000	A BIRD IN THE HAND IS WORTH TWO IN THE BUSH	A BIRD IN THE HAND IS WORTH TWO IN THE BUSH
mediapipe	4.63	0.03	1.83	40.04	0.500	0.268	THE ROBOT CANT MAKE AN OMELET WITHOUT BREAKING SOME EGGS	THE ROBOT CAN MAKE AN OBJECT WITHOUT BRINGING DEMANDS
mediapipe	3.30	0.03	1.33	25.87	0.125	0.118	YOU CANT JUDGE A BOOK BY ITS COVER	YOU CANT CHANGE A BOOK BY ITS COVER
mediapipe	2.67	0.03	1.23	15.84	0.167	0.125	HIT THE NAIL ON THE HEAD	AND THE NAIL ON THE HEAD
mediapipe	2.80	0.03	1.10	16.70	0.000	0.000	ACTIONS SPEAK LOUDER THAN WORDS	ACTIONS SPEAK LOUDER THAN WORDS
mediapipe	3.03	0.03	1.17	20.40	0.714	0.379	YOUR GUESS IS AS GOOD AS MINE	YOUR GUEST IS A JOURNALIST
mediapipe	3.57	0.03	1.37	38.57	0.222	0.049	WELL CROSS THAT BRIDGE WHEN WE COME TO IT	WE ALL CROSS THAT BRIDGE WHEN WE COME TO IT
mediapipe	3.27	0.03	1.40	23.50	0.333	0.313	TIME FLIES WHEN YOURE HAVING FUN	STOP WHEN YOURE HAVING FUN
mediapipe	3.27	0.03	1.30	21.87	0.375	0.182	TAKE A LONG WALK OFF A SHORT PIER	TAKE A LONG WALK OF THE SHORT PAIR
mediapipe	3.17	0.03	2.40	32.14	0.667	0.720	HE HAS BIGGER FISH TO FRY	HE HAS BEEN OFFICIALLY DEFINED
mediapipe	3.10	0.03	1.90	23.37	0.000	0.000	GOOD THINGS COME TO THOSE WHO WAIT	GOOD THINGS COME TO THOSE WHO WAIT
mediapipe	3.27	0.03	1.77	20.40	0.000	0.000	A PICTURE IS WORTH A THOUSAND WORDS	A PICTURE IS WORTH A THOUSAND WORDS
mediapipe	3.30	0.07	1.14	168.77	0.000	0.000	DONT COUNT YOUR CHICKENS BEFORE THEY HATCH	DONT COUNT YOUR CHICKENS BEFORE THEY HATCH
retina_fan	4.17	0.03	1.60	207.50	0.091	0.041	DO UNTO OTHERS AS YOU WOULD HAVE THEM DO UNTO YOU	DO UNTO OTHERS AS YOU WOULD HAVE THEM DO TO YOU
retina_fan	2.97	0.03	1.27	128.03	0.000	0.000	THE DEVIL IS IN THE DETAILS	THE DEVIL IS IN THE DETAILS
retina_fan	3.63	0.03	1.43	156.64	0.750	0.611	IT AINT OVER TILL THE FAT LADY SINGS	AND HEAD OVER TO THE VIOLIN HE SAYS
retina_fan	3.67	0.03	1.40	164.74	0.000	0.000	A BIRD IN THE HAND IS WORTH TWO IN THE BUSH	A BIRD IN THE HAND IS WORTH TWO IN THE BUSH
retina_fan	3.07	0.03	1.30	131.70	0.000	0.000	A PENNY SAVED IS A PENNY EARNED	A PENNY SAVED IS A PENNY EARNED
retina_fan	3.60	0.03	1.37	156.07	0.714	0.393	HES A CHIP OFF THE OLD BLOCK	IS THE SHIP OF THE NORTH BLOCK
retina_fan	2.80	0.03	1.20	122.04	0.200	0.161	ACTIONS SPEAK LOUDER THAN WORDS	ACTUALLY SPEAK LOUDER THAN WORDS
retina_fan	2.87	0.03	1.20	125.70	1.333	0.778	IGNORANCE IS BLISS	ACCORDING TO HIS PLACE
retina_fan	2.43	0.03	1.13	105.14	1.000	0.600	NO PAIN NO GAIN	YOU PAN YOUR HAND
retina_fan	2.80	0.03	1.80	127.50	0.667	0.607	THE EARLY BIRD GETS THE WORM	THE HALL APPEARED AGAINST THE WALL
retina_fan	3.27	0.03	1.63	144.50	0.000	0.000	A PICTURE IS WORTH A THOUSAND WORDS	A PICTURE IS WORTH A THOUSAND WORDS
retina_fan	3.10	0.03	1.27	146.57	0.000	0.000	GOOD THINGS COME TO THOSE WHO WAIT	GOOD THINGS COME TO THOSE WHO WAIT
retina_fan	3.30	0.03	1.40	165.30	0.125	0.118	YOU CANT JUDGE A BOOK BY ITS COVER	YOU CANT CHANGE A BOOK BY ITS COVER
retina_fan	3.27	0.03	1.40	149.97	0.250	0.152	TAKE A LONG WALK OFF A SHORT PIER	TAKE A LONG WALK OFF THE SHORT PAIR
retina_fan	2.67	0.03	1.17	117.00	0.333	0.208	HIT THE NAIL ON THE HEAD	AND THE NAIL ON THE HAND
retina_fan	3.03	0.03	1.23	135.27	0.143	0.034	YOUR GUESS IS AS GOOD AS MINE	YOUR GUEST IS AS GOOD AS MINE
retina_fan	3.34	0.03	1.30	145.79	0.000	0.000	SLOW AND STEADY WINS THE RACE	SLOW AND STEADY WINS THE RACE
retina_fan	2.27	0.03	1.03	97.30	0.000	0.000	LIVE AND LEARN	LIVE AND LEARN
retina_fan	3.57	0.03	1.53	175.27	0.222	0.049	WELL CROSS THAT BRIDGE WHEN WE COME TO IT	WE ALL CROSS THAT BRIDGE WHEN WE COME TO IT
retina_fan	3.03	0.03	1.40	138.70	0.250	0.222	HOW MUCH IS THAT DOGGY IN THE WINDOW	HOW MUCH IS THAT TALKING IN THE WINDOWS
retina_fan	3.17	0.03	1.27	137.70	0.800	0.400	ITS RAINING CATS AND DOGS	ITS READING AS ADULTS
retina_fan	2.87	0.03	1.20	122.84	0.200	0.158	ITS A PIECE OF CAKE	ITS A PIECE OF AIR
retina_fan	2.47	0.03	1.23	119.73	0.500	0.462	MAKE LIKE A TREE AND LEAVE	BACK ON A JOURNEY AND LEAVE
retina_fan	3.17	0.03	1.33	168.37	0.667	0.720	HE HAS BIGGER FISH TO FRY	HE HAS BEEN OFFICIALLY DEFINED
retina_fan	4.63	0.03	1.70	216.80	0.400	0.214	THE ROBOT CANT MAKE AN OMELET WITHOUT BREAKING SOME EGGS	THE ROBOT CAN MAKE AN OBJECT WITHOUT BRINGING SOME HANDS
retina_fan	2.47	0.03	1.23	109.70	0.000	0.000	LOOK BEFORE YOU LEAP	LOOK BEFORE YOU LEAP
retina_fan	2.27	0.03	1.10	104.84	0.500	0.227	BETTER LATE THAN NEVER	BETTER LIGHT THAN EVER
retina_fan	3.27	0.03	1.33	143.30	0.333	0.313	TIME FLIES WHEN YOURE HAVING FUN	STOP WHEN YOURE HAVING FUN
retina_fan	3.13	0.03	1.23	138.97	0.000	0.000	EVERY CLOUD HAS A SILVER LINING	EVERY CLOUD HAS A SILVER LINING

Table 11: Results obtained for participant number 4 on the laptop.

B.2 Retina FAN vs MediaPipe on server

Participant	Recording [s]	Transferring [s]	Video Building [s]	Video Processing [s]	WER	CER
1	3.91	0.03	0.61	8.406	0.286	0.195
2	3.95	0.03	0.61	6.671	0.321	0.197
3	3.13	0.04	0.54	5.491	0.352	0.220
8	3.64	0.04	0.61	6.380	0.363	0.234
4	3.38	0.03	0.55	5.932	0.462	0.291
9	4.42	0.03	0.69	7.605	0.542	0.367
10	3.55	0.04	0.60	6.227	0.617	0.394
5	3.62	0.03	0.60	6.493	0.635	0.449
11	2.94	0.03	0.54	5.439	0.706	0.522
12	4.09	0.03	0.61	7.036	0.719	0.520
6	2.97	0.03	0.53	5.366	0.722	0.515
13	2.96	0.03	0.54	5.505	0.785	0.585
14	4.23	0.03	0.68	7.514	0.787	0.558
7	2.83	0.03	0.53	5.407	0.795	0.533
15	4.37	0.06	0.66	7.885	0.796	0.593
16	2.88	0.03	0.53	5.449	0.814	0.604
17	2.79	0.03	0.52	5.476	0.862	0.651
18	3.32	0.03	0.55	6.143	0.896	0.678
19	4.96	0.04	0.68	9.103	1.016	0.865
AVERAGE	3.58 ± 0.64	0.04 ± 0.01	0.59 ± 0.06	6.502 ± 1.135	0.657 ± 0.213	0.472 ± 0.185

Table 12: Results obtained using Retina FAN on the server. The green rows are participants that are English native speaker.

Participant	Recording [s]	Transferring [s]	Video Building [s]	Video Processing [s]	WER	CER
1	3.91	0.04	0.63	6.535	0.275	0.171
2	3.95	0.03	0.63	2.421	0.311	0.186
3	3.12	0.05	0.57	2.254	0.337	0.216
8	3.64	0.05	0.62	2.47	0.353	0.219
9	4.43	0.04	0.70	2.734	0.495	0.318
5	3.61	0.03	0.63	2.633	0.65	0.469
11	2.93	0.03	0.53	2.306	0.655	0.45
10	3.55	0.03	0.63	2.404	0.673	0.457
12	4.09	0.03	0.63	2.718	0.682	0.482
6	2.97	0.03	0.54	2.179	0.696	0.493
13	4.23	0.06	0.75	2.989	0.766	0.531
7	2.83	0.04	0.54	4.468	0.774	0.544
14	4.38	0.04	0.64	3.038	0.811	0.613
15	2.96	0.03	0.53	2.406	0.812	0.594
16	2.88	0.03	0.53	4.502	0.84	0.62
17	2.79	0.03	0.52	2.501	0.877	0.676
18	3.32	0.02	0.56	2.689	0.911	0.692
19	4.99	0.04	0.68	3.68	1.054	0.846
4	3.38	0.05	0.56	7.004	1.315	1.017
AVERAGE	3.58 ± 0.64	0.04 ± 0.01	0.60 ± 0.06	3.259 ± 1.410	0.699 ± 0.265	0.505 ± 0.223

Table 13: Results obtained using MediaPipe on the server. The green rows are participants that are English native speaker.

B.3 Face's Landmarks Detection Fps Comparison

	Laptop	**Server**
MediaPipe	130	200
Retina FAN	<1	23

Table 14: Fps comparison.

C Rqt Graphs

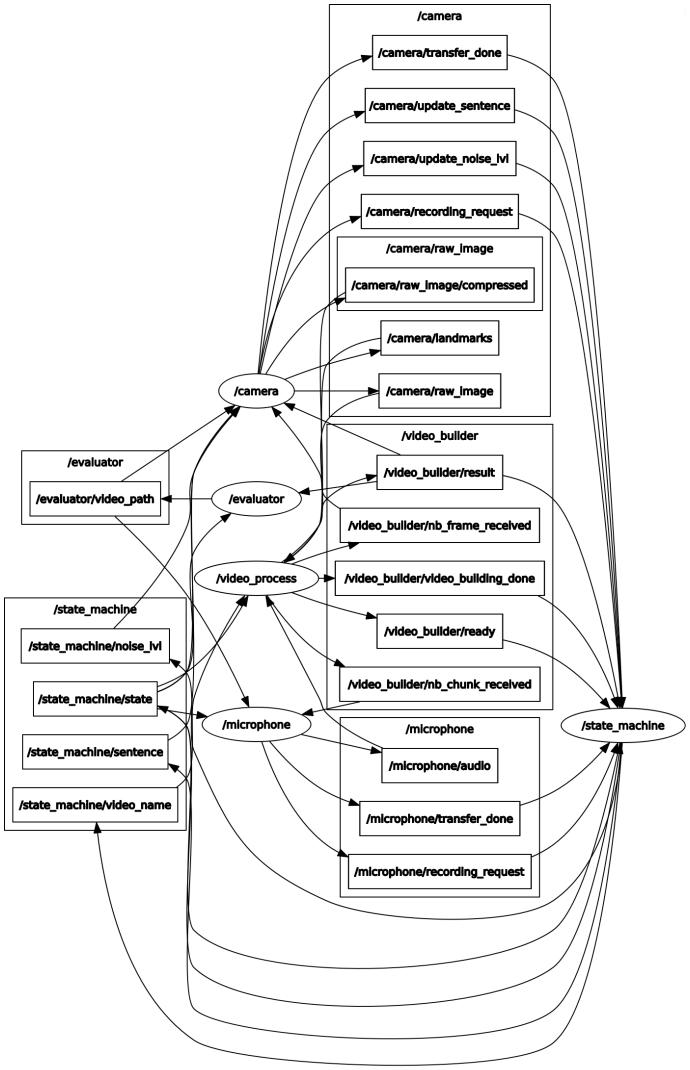

Figure 4: This is the rqt graph containing all the nodes of the systems listed in section 4. The nodes are circled and the topics are in rectangles. An arrow starting from a node to a topic means that the node is a publisher of that topic. An arrow starting from a topic to a node means that the node is a subscriber to that topic. All topics' names start by /PUBLISHER_NODE_NAME/ and are then regrouped in boxes by publisher. The decomposition of the graph per nodes can be seen in sections 4.1 to 4.5

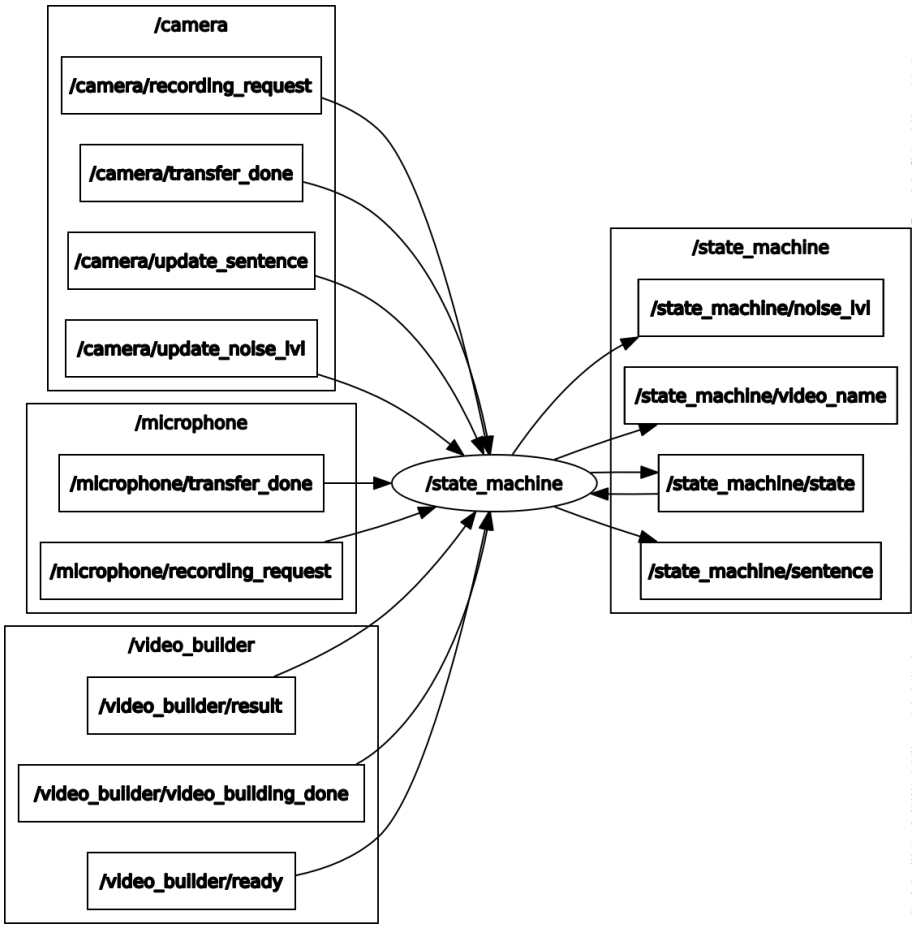

Figure 5: This is the rqt graph of the State Manager. See Figure 4 captions for the rqt graph's key.

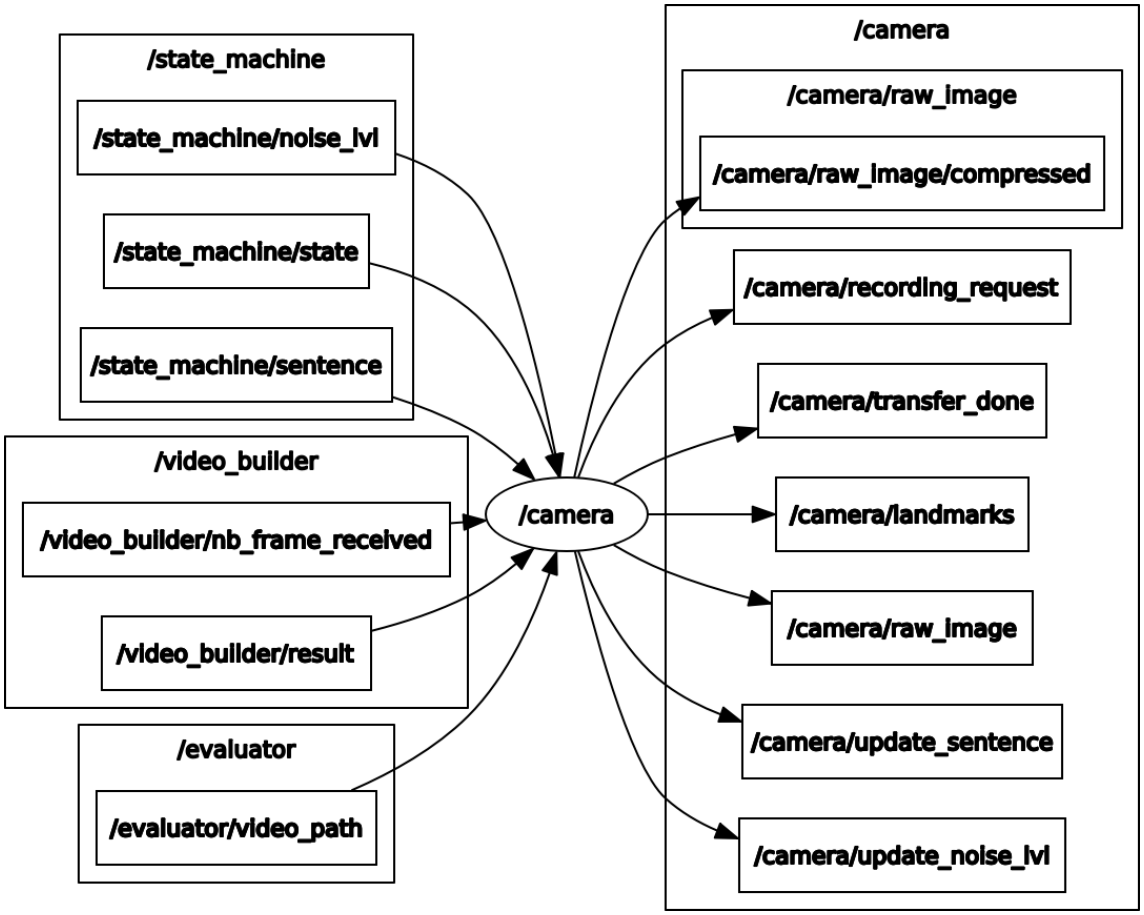

Figure 6: This is the rqt graph of the Visual. See Figure 4 captions for the rqt graph's key.

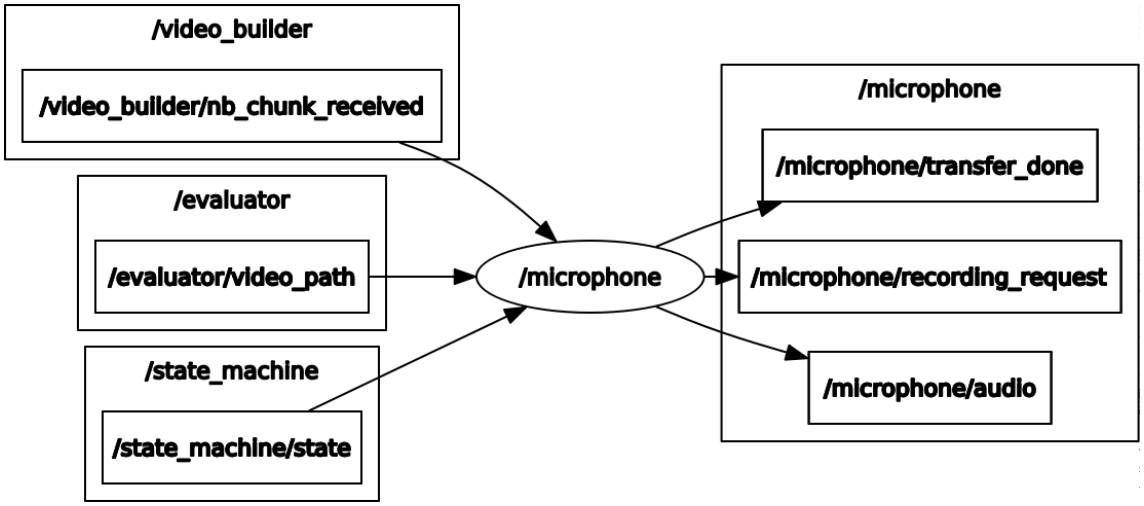

Figure 7: This is the rqt graph of the Audio. See Figure 4 captions for the rqt graph's key.

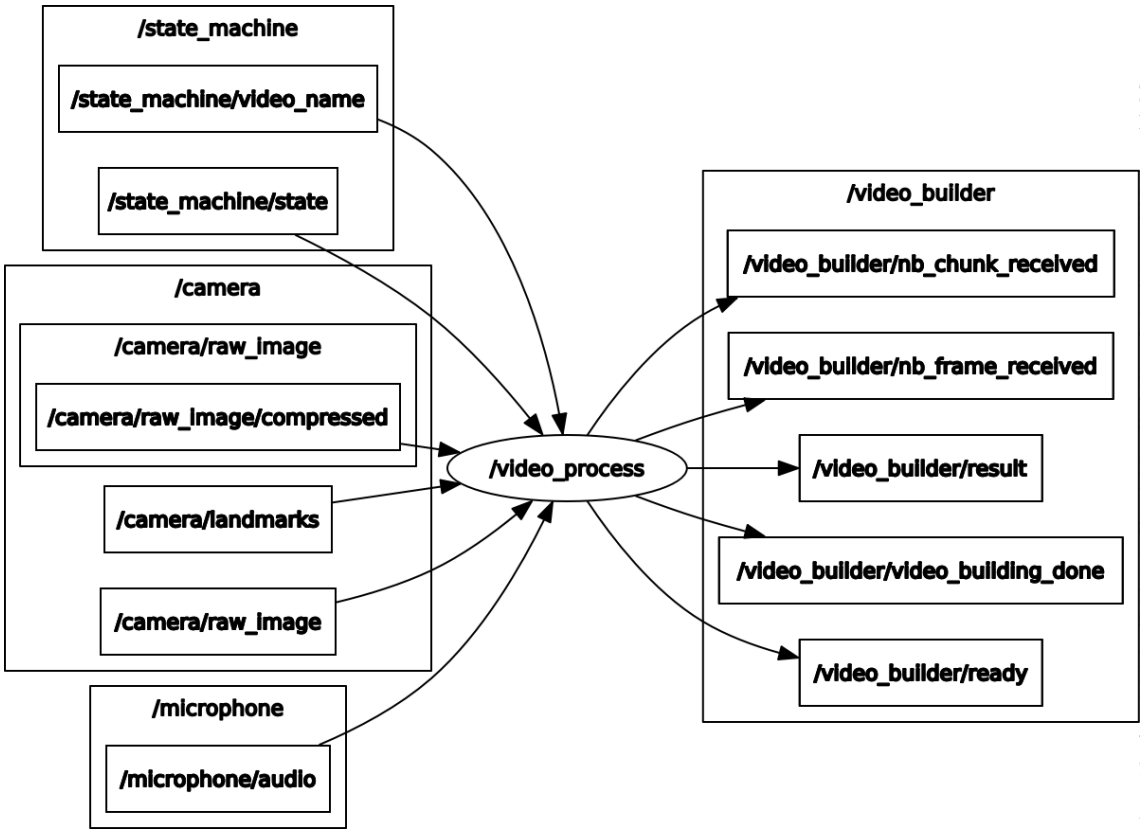

Figure 8: This is the rqt graph of the Video Module. See Figure 4 captions for the rqt graph's key.

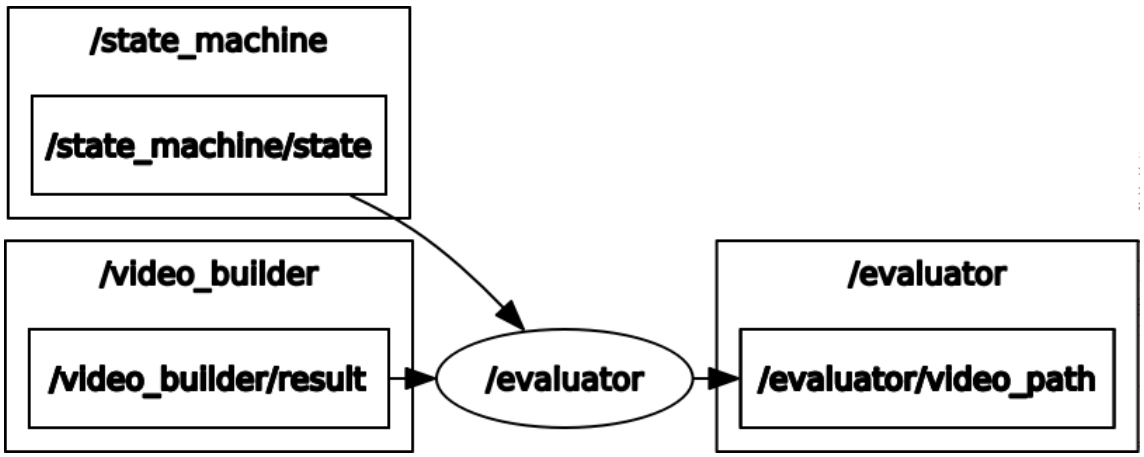

Figure 9: This is the rqt graph of the Evaluator. See Figure 4 captions for the rqt graph's key.