Title: Speech Recognition - Automatic Lip Reading Model using 3D CNN and GRU

Team Name: EDITH

Team Members:

1. Yarlagadda Sai Manoj
2. S R Suhas

General Description:

The project focuses on building an automatic lip reading model using deep learning techniques, specifically 3D Convolutional Neural Networks (CNN) and Gated Recurrent Unit (GRU). Lip reading is the process of interpreting speech solely through visual cues, which has significant potential as a complementary modality to speech recognition or as a standalone visual speech recognition system. This research aims to develop a lip reading model for the Indonesian language, which has been limited in prior studies, and make it applicable in real-world scenarios by supporting variable-length sentences as input.

PREREQUISITE:

Creating a Speech Recognition - Automatic Lip Reading Model using 3D CNN and GRU involves combining several technologies and concepts. Here are some prerequisites that would be beneficial to understand before diving into this project:

Deep Learning Fundamentals:

A solid understanding of deep learning concepts, including neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and gated recurrent units (GRUs).

Machine Learning and Data Preprocessing:

Familiarity with machine learning principles, data preprocessing, and feature extraction. You'll need to preprocess both the audio and video data to make them suitable for training the model.

Speech Recognition:

A good grasp of speech recognition techniques and methodologies. This includes understanding how to process audio signals, convert them into spectrograms or other relevant representations, and handle the temporal aspect of speech.

Computer Vision and 3D CNNs:

Understanding of computer vision concepts and 3D convolutional neural networks. 3D CNNs are used to process sequential data, like video frames, and can capture both spatial and temporal features.

Lip Reading and Lip Movements:

Knowledge of lip reading, including the movements and patterns associated with different phonemes and words. This will help you design the architecture and loss functions to capture the relevant features.

RNNs and GRUs:

Familiarity with recurrent neural networks (RNNs) and gated recurrent units (GRUs). These are essential for modeling the temporal dependencies in the lip movement sequences.

Python and Deep Learning Libraries:

Proficiency in Python programming is a must. You'll also need to work with deep learning libraries like TensorFlow or PyTorch to implement your model.

Data Collection and Annotation:

Gathering a suitable dataset of paired audio and video samples where the speech is clearly visible on the lips. Annotation of these data samples is also crucial for supervised learning.

Data Augmentation:

Understanding of data augmentation techniques to artificially increase the diversity of your training data. This can help improve the generalization of your model.

Loss Functions:

Knowledge of appropriate loss functions for multi-modal data (audio and video). You may need to design custom loss functions that consider both the audio and visual modalities.
Model Evaluation:

Understanding how to evaluate the performance of your model using metrics such as accuracy, precision, recall, and F1-score. You might also consider using techniques like cross-validation.

GPU and Parallel Computing:

Familiarity with GPU programming and parallel computing is advantageous, as training deep learning models can be computationally intensive. GPUs can significantly speed up the training process.

Model Optimization:

Techniques for optimizing your model, including hyperparameter tuning, regularization, and techniques for reducing overfitting.

Research Papers and Resources:

Familiarity with relevant research papers and resources related to speech recognition, lip reading, and multimodal learning. This will help you stay updated on the latest advancements and best practices in the field.

Remember that working on a complex project like this may require continuous learning and experimentation. Start with smaller, simpler projects to build your understanding and then gradually work your way up to more complex tasks like the one you've described.

STEPS TO IMPLEMENT THE MODEL:

Implementing a Speech Recognition - Automatic Lip Reading Model using 3D CNN and GRU involves several steps. Here's a general outline of the process:

1. **Data Collection and Preprocessing:**

Collect a dataset of paired audio and video samples where the speech is clearly visible on the lips. Preprocess the audio by converting it into spectrograms or other relevant audio representations. Preprocess the video frames by extracting the lip regions or using pre-trained face detection models.

2. **Data Augmentation:**

   Apply data augmentation techniques to increase the diversity of your training data. This can include random cropping, flipping, and adding noise to the data.

3. **Model Architecture Design:**

   Design the architecture of your model. You'll likely have a 3D CNN component for processing the video frames and a GRU component for capturing temporal dependencies. You might also have separate branches for processing audio and video data.

4. **Loss Function Design:**

   Design a suitable loss function that takes into account both the audio and visual modalities. This could involve creating custom loss functions or combining existing ones.

5. **Model Implementation:**

   Implement your model using a deep learning library like TensorFlow or PyTorch. Define the layers, connections, and flow of data through the model.

6. **Training:**

   Split your dataset into training, validation, and possibly test sets. Train your model using the training data and validate its performance using the validation set. Monitor metrics like loss and accuracy during training.

7. **Hyperparameter Tuning:**

   Tune hyperparameters such as learning rate, batch size, and regularization parameters to optimize your model's performance. You might use techniques like grid search or random search.
8. **Evaluation:**

Evaluate your trained model on the test set to assess its generalization performance. Use appropriate metrics for speech recognition and lip reading tasks.

9. **Fine-Tuning and Iteration:**

   Analyze the model's performance and make necessary adjustments. This could involve changing the architecture, modifying hyperparameters, or introducing new techniques.

10. **Deployment:**

    Once you're satisfied with your model's performance, you can deploy it for real-world applications. This might involve integrating it into a larger system or creating a user-friendly interface.

11. **Documentation:**

    Document your model architecture, preprocessing steps, hyperparameters, and other relevant details. This documentation will be useful for future reference and for sharing your work with others.

12. **Continued Learning and Improvement:**

    Stay updated with the latest research and techniques in the fields of speech recognition, lip reading, and deep learning. As the field evolves, you might find opportunities to improve your model's performance.

Remember that each step in the process requires careful attention and experimentation. It's common to iterate on your model and its components to achieve the best possible results. Additionally, don't hesitate to consult relevant research papers, online tutorials, and forums to get insights and help along the way.

BUILD THE SOLUTION:

Building an automatic lip reading model using a combination of 3D Convolutional Neural Networks (CNNs) and Gated Recurrent Units (GRUs) involves several steps. This type of model is designed to take in video frames of lip movements and predict the corresponding spoken words. Here's a high-level overview of the process:

Data Collection and Preparation:

Collect a dataset of videos where the spoken words are synchronized with the video frames. This dataset should include both the video frames and their corresponding transcriptions.

Data Preprocessing:

Extract video frames from the videos and convert them to a consistent format (e.g., grayscale frames).
Preprocess the transcriptions, which might involve tokenization, converting words to numerical representations, and padding sequences to a fixed length.
Model Architecture:
Design the architecture of the automatic lip reading model, which consists of a 3D CNN followed by a GRU network.

3D CNN:

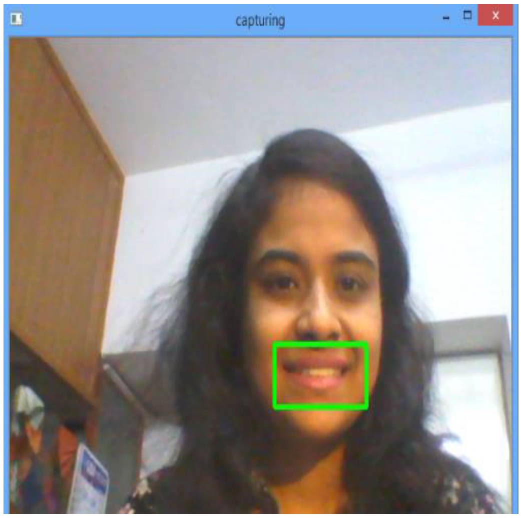
The 3D CNN will learn spatial and temporal features from the video frames. It processes both the spatial dimensions (width and height of the frames) and the temporal dimension (frames over time).

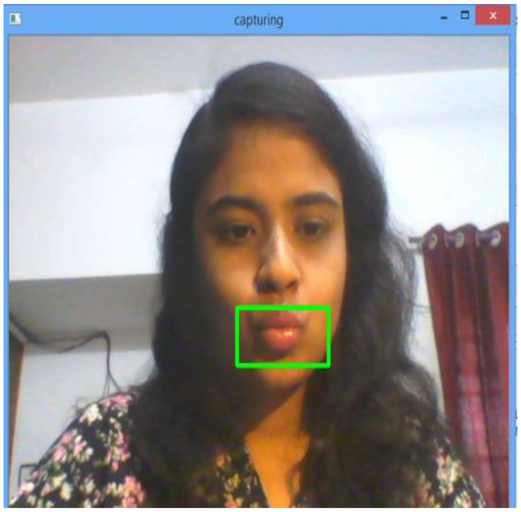
GRU Network:

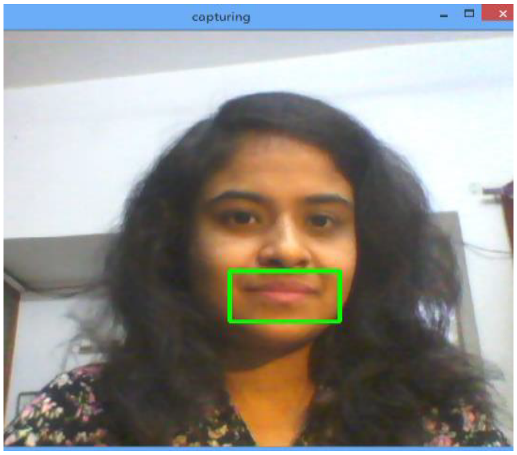
The GRU network takes the output from the 3D CNN as sequential features and learns the temporal dependencies within the sequence. It captures the context and context changes over time.

Model Implementation:

Implement the model architecture using a deep learning framework such as TensorFlow or PyTorch.



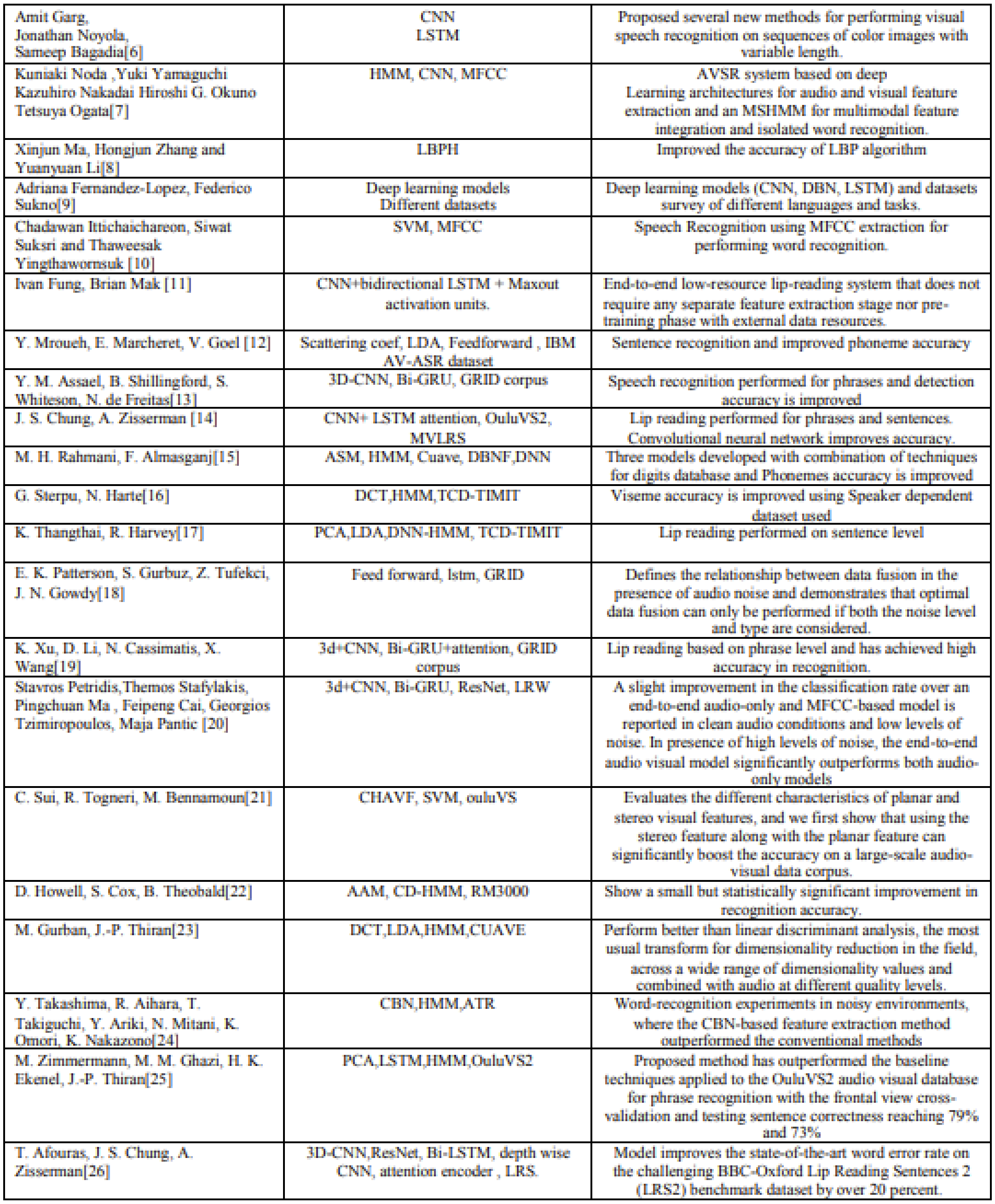




Literature survey of Few English datasets are listed below:

Table 2: Overview of few English Databases.

Name	Year	Language	Speaker	Task	utterances	Duration
AVLetters[27]	1998	English	10	Alphabet	780	13 min
XM2VTS[28]	1999	English	295	Digits	885	59 min
IBMVIAVOICE[29]	2000	English	290	Sentences	10,500	50 h
CUAVE[30]	2004	English	36	Digits	7,000	14 min
GRID[31]	2006	English	34	Phrases	34,000	28 h
Ouluvs[32]	2009	English	150	Sentences	(N/A)	20 h
MOBIO[33]	2012	English	150	Sentences	(N/A)	20 h
AusTalk[34]	2014	English	1000	Digits Words Phrases	24,000 966000 59000	3000 h
LRS[4]	2017	English	29	Sentences	118,166	33 h
AVDigits[35]	2018	English	10	Digits Phrases	795 5850	(N/A)



Amit Garg, Jonathan Noyola, Sameep Bagadia[6]	CNN LSTM	Proposed several new methods for performing visual speech recognition on sequences of color images with variable length.
Kuniaki Noda ,Yuki Yamaguchi Kazuhiro Nakadai Hiroshi G. Okuno Tetsuya Ogata[7]	HMM, CNN, MFCC	AVSR system based on deep Learning architectures for audio and visual feature extraction and an MSHMM for multimodal feature integration and isolated word recognition.
Xinjun Ma, Hongjun Zhang and Yuanyuan Li[8]	LBPH	Improved the accuracy of LBP algorithm
Adriana Fernandez-Lopez, Federico Sukno[9]	Deep learning models Different datasets	Deep learning models (CNN, DBN, LSTM) and datasets survey of different languages and tasks.
Chadawan Ittichaichareon, Siwat Suksri and Thaweesak Yingthawornsuk [10]	SVM, MFCC	Speech Recognition using MFCC extraction for performing word recognition.
Ivan Fung, Brian Mak [11]	CNN+bidirectional LSTM + Maxout activation units.	End-to-end low-resource lip-reading system that does not require any separate feature extraction stage nor pre-training phase with external data resources.
Y. Mroueh, E. Marcheret, V. Goel [12]	Scattering coef, LDA, Feedforward , IBM AV-ASR dataset	Sentence recognition and improved phoneme accuracy
Y. M. Assael, B. Shillingford, S. Whiteson, N. de Freitas[13]	3D-CNN, Bi-GRU, GRID corpus	Speech recognition performed for phrases and detection accuracy is improved
J. S. Chung, A. Zisserman [14]	CNN+ LSTM attention, OuluVS2, MVLRS	Lip reading performed for phrases and sentences. Convolutional neural network improves accuracy.
M. H. Rahmani, F. Almasganj[15]	ASM, HMM, Cuave, DBNF,DNN	Three models developed with combination of techniques for digits database and Phonemes accuracy is improved
G. Sterpu, N. Harte[16]	DCT,HMM,TCD-TIMIT	Viseme accuracy is improved using Speaker dependent dataset used
K. Thangthai, R. Harvey[17]	PCA,LDA,DNN-HMM, TCD-TIMIT	Lip reading performed on sentence level
E. K. Patterson, S. Gurbuz, Z. Tufekci, J. N. Gowdy[18]	Feed forward, lstm, GRID	Defines the relationship between data fusion in the presence of audio noise and demonstrates that optimal data fusion can only be performed if both the noise level and type are considered.
K. Xu, D. Li, N. Cassimatis, X. Wang[19]	3d+CNN, Bi-GRU+attention, GRID corpus	Lip reading based on phrase level and has achieved high accuracy in recognition.
Stavros Petridis,Themos Stafylakis, Pingchuan Ma , Feipeng Cai, Georgios Tzimiropoulos, Maja Pantic [20]	3d+CNN, Bi-GRU, ResNet, LRW	A slight improvement in the classification rate over an end-to-end audio-only and MFCC-based model is reported in clean audio conditions and low levels of noise. In presence of high levels of noise, the end-to-end audio visual model significantly outperforms both audio-only models
C. Sui, R. Togneri, M. Bennamoun[21]	CHAVF, SVM, ouluVS	Evaluates the different characteristics of planar and stereo visual features, and we first show that using the stereo feature along with the planar feature can significantly boost the accuracy on a large-scale audio-visual data corpus.
D. Howell, S. Cox, B. Theobald[22]	AAM, CD-HMM, RM3000	Show a small but statistically significant improvement in recognition accuracy.
M. Gurban, J.-P. Thiran[23]	DCT,LDA,HMM,CUAVE	Perform better than linear discriminant analysis, the most usual transform for dimensionality reduction in the field, across a wide range of dimensionality values and combined with audio at different quality levels.
Y. Takashima, R. Aihara, T. Takiguchi, Y. Ariki, N. Mitani, K. Omori, K. Nakazono[24]	CBN,HMM,ATR	Word-recognition experiments in noisy environments, where the CBN-based feature extraction method outperformed the conventional methods
M. Zimmermann, M. M. Ghazi, H. K. Ekenel, J.-P. Thiran[25]	PCA,LSTM,HMM,OuluVS2	Proposed method has outperformed the baseline techniques applied to the OuluVS2 audio visual database for phrase recognition with the frontal view cross-validation and testing sentence correctness reaching 79% and 73%
T. Afouras, J. S. Chung, A. Zisserman[26]	3D-CNN,ResNet, Bi-LSTM, depth wise CNN, attention encoder , LRS.	Model improves the state-of-the-art word error rate on the challenging BBC-Oxford Lip Reading Sentences 2 (LRS2) benchmark dataset by over 20 percent.

Python Code:

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv3D, MaxPooling3D, Flatten, GRU, Dense

model = Sequential([
    Conv3D(filters=32, kernel_size=(3, 3, 3), activation='relu', input_shape=(frames, height, width, channels)),
    MaxPooling3D(pool_size=(2, 2, 2)),
    Flatten(),
    GRU(units=128, return_sequences=True),
    Dense(vocab_size, activation='softmax')
])
```

Model Training:

Split your dataset into training and validation sets.
Define a loss function, typically categorical cross-entropy, and an optimizer like Adam.
Train the model using the preprocessed video frames as input and transcriptions as target outputs.

Model Evaluation:

Evaluate the model's performance on a separate test dataset. Calculate metrics such as accuracy, precision, recall, and F1-score. Adjust the model hyperparameters as needed.

Inference:

Once the model is trained and evaluated, you can use it for inference. Pass new video frames through the model to generate predictions (transcriptions).

Deployment:

Depending on your use case, deploy the model to production. This might involve creating an API, integrating the model into an application, or running it on edge devices.

Remember that building a high-performing lip reading model requires careful tuning of hyperparameters, architecture choices, and significant amounts of high-quality data. Additionally, keep track of the latest research and advancements in the field to ensure your model remains competitive.

Novelty/Uniqueness:

The uniqueness of this project lies in several aspects. Firstly, it addresses the limited research on lip reading for the Indonesian language, making it one of the pioneering efforts in this domain. Additionally, the model's capability to handle variable-length sentences is a significant advancement, allowing it to process sentences of different lengths in real-world conversations. The use of 3D CNN enables the model to capture both spatial and temporal information from video frames, while the GRU serves as a character-level sentence decoder, enhancing the accuracy of lip reading predictions. Furthermore, the investigation of whether knowledge from lip reading in other languages affects the acquisition of a different language introduces a cross-lingual perspective to the study.

Business/Social Impact:

The developed automatic lip reading model holds promising applications across various domains. As a complementary modality to speech recognition systems, it can enhance speech-to-text accuracy in noisy environments or situations where audio-based recognition faces challenges. Furthermore, the model's ability to perform visual speech recognition opens up possibilities for silent speech interfaces, enabling communication in noise-sensitive environments, aiding individuals with speech impairments, and supporting hands-free communication in crowded places. Additionally, the research contributes to the development of language-specific lip reading technology, which can be extended to other languages beyond Indonesian, benefiting diverse linguistic communities.

Technology Stack:

The project utilizes deep learning techniques to build the lip reading model. The technology stack includes:

1. Python: The primary programming language for implementing the model and related tasks.
2. TensorFlow/Keras: Used for creating and training the 3D CNN and GRU models.
3. 3D Convolutional Neural Network (CNN): Responsible for extracting spatial and temporal features from video frames.
4. Gated Recurrent Unit (GRU): Functions as the character-level sentence decoder for lip reading predictions.
5. Data Preprocessing: Techniques for extracting frames from videos, aligning frames with corresponding labels, and handling variable-length sentences.

Scope of the Work:

The project's scope involves building and training the automatic lip reading model using 3D CNN and GRU for the Indonesian language. The research will focus on collecting a suitable dataset of video samples with corresponding transcriptions or labels. Data preprocessing will involve extracting and aligning frames, as well as handling sentences of varying lengths.

The model will be implemented using TensorFlow/Keras, and its performance will be evaluated using appropriate metrics. The research will explore the effectiveness of the 3D CNN and GRU architecture for lip reading tasks and assess the model's ability to handle variable-length input.

Additionally, the project will investigate the transfer learning potential of knowledge acquired from lip reading in other languages to improve the performance of the Indonesian language lip reading model.

The ultimate goal is to develop a robust lip reading model that can be applied in real-world scenarios, supporting practical applications such as speech recognition enhancements, silent speech interfaces, and communication assistance for individuals with speech impairments. The potential for extending the research to other languages and diverse applications adds value to the significance of the work.

Conclusion:

In conclusion, building a Silent Speech Recognition system using a 3D CNN and GRU is a complex and challenging endeavor, but it holds great promise for revolutionizing speech recognition technology. The project combines computer vision and deep learning techniques to extract and analyze lip movements from video sequences, enabling speech understanding without relying on audio input.

The development of a Silent Speech Recognition system using 3D CNN and GRU represents a remarkable advancement in the field of speech recognition and has numerous potential applications, ranging from improving human-computer interactions to assisting individuals with speech disabilities. However, it remains an ongoing area of research and requires continuous innovation and refinement to achieve its full potential. With the right expertise, resources, and dedication, this technology can significantly contribute to enhancing communication and accessibility in diverse real-world scenarios.

We have discussed various deep learning, machine learning techniques and approaches for lip reading. As well as we discussed various types of available datasets. Deep learning can classify, cluster, and predict anything id we have data like images, videos, sound, text etc. It is observed that lip reading systems are currently dominated by CNN features in combination with

LSTM. It has provided significant improvement in terms of performance. Different types of datasets are available like character, word, sentence, digits and phrase. The datasets are also available in various languages English, French, German, Japanese etc. Datasets for Indian languages can also be prepared. In this survey we can observe that datasets are only available in few languages we can create a datasets for a regional languages and can thus contribute to the society. In India 70% people live in a rural area so for them regional database should be created and thus taking technology to the remote areas. This gave us the brief idea about the Deep learning approaches and which approach can yield good results

Reference:

https://www.kaggle.com/datasets/apoorvwatsky/miraclvc1