Silent Help: Arabic Lip Reading Emergency System (SHARES)

Yasser Alharbi, Sultan Alaqili, Nawaf Alandijany, Abdulaziz AbuTaleb, Naif Alharbi

yasser.luq@outlook.com sultanalothale22@gmail.com nwandijany@gmail.com abdulaziz.h.abutaleb@hotmail.com NaifMutebAlharbi@gmail.com

Department of Computer Science and Artificial Intelligence
Umm Al-Qura University, Saudi Arabia

Abstract

This paper presents **SHARES** (Silent Help: Arabic Lip Reading Emergency System), an innovative Arabic lip-reading system designed to recognize spoken Arabic words through visual inputs. The system leverages advanced deep learning architectures, specifically the R(2+1)D convolutional neural network (CNN), fine-tuned on a proprietary Arabic dataset. SHARES focuses on recognizing common Arabic words used in emergency scenarios, such as اسعاف ("Ambulance") and ساعدني ("Help me"). By addressing the challenges posed by Arabic visual speech patterns, SHARES bridges the communication gap for individuals with hearing or speech impairments. Experimental results demonstrate that SHARES achieves a **76% accuracy** on the test set, highlighting its effectiveness in real-time emergency communication. This paper provides a comprehensive overview of SHARES, including the dataset creation process, preprocessing pipeline, model design, and evaluation methodology.

Keywords: Arabic, Computer Vision, CNN, Emergency, Fine-Tuning.

1 Introduction

Lip-reading, or visual speech recognition, involves interpreting spoken words by analyzing lip movements. It has significant applications in assistive technologies for individuals with hearing impairments, communication in noisy environments, and advanced human-computer interaction. In emergency situations, rapid and precise communication is critical, making lip-reading technology an indispensable tool for accessibility. These systems enable individuals who cannot hear or speak to communicate effectively, ensuring timely assistance and enhancing safety.

Despite progress in lip-reading models for languages like English, Arabic remains an underexplored domain due to the lack of annotated datasets and the complexity of Arabic phonetics and script. Furthermore, existing systems often face challenges related to accuracy and speed in recognizing context-specific vocabulary in real time. However, recent advancements in machine learning and image processing have paved the way for more sophisticated Arabic lip-reading models.

SHARES (Silent Help: Arabic Lip Reading Emergency System) addresses this gap by focusing on real-time recognition of essential Arabic words in emergency scenarios. It uses the R (2 + 1) D CNN architecture, known for its ability to extract both spatial and temporal features, to improve recognition accuracy. This paper presents the development of SHARES, detailing its dataset construction, preprocessing techniques, and model design, while highlighting its potential to bridge communication gaps and improve emergency response effectiveness.

Figure 1: Example of mouth-closed detection. The system accurately identifies the closed-mouth state.

Figure 2: Example of mouth-open detection. The system successfully detects when the mouth is open, critical for recognizing the emergency Arabic word.

Figure 3: Real-time bounding box overlay and Arabic text display. This example highlights the system's ability to process input in real-time, with accurate predictions displayed of the emergency Arabic word حريق.

2 Literature Review

2.1 Computer Vision in Lip Reading

Lip reading, or visual speech recognition, has garnered significant attention in the field of computer vision due to its potential applications in assistive technologies and human-computer interaction. A notable project in this domain is the work conducted by Ye and Kwong, who developed a speech recognition system that uses computer vision and deep learning techniques to accurately recognize spoken words from visual input (Ye and Kwong, 2023).

The authors aimed to create a system capable of recognizing a predefined set of spoken words through visual analysis of lip movements. The project involved the construction of a substantial dataset, comprising approximately 700 video clips of individuals articulating individual words, resulting in a data set size of approximately 3 GB. The videos were manually labeled and encompassed 13 different words, ensuring a balanced distribution of samples across classes.

The system architecture integrated several convolutional and dense layers within a 3D Convolutional Neural Network (3D CNN) framework. Utilizing libraries such as TensorFlow and Keras for model training, and OpenCV and PIL for data preprocessing, the authors emphasized the importance of image processing techniques in enhancing model performance. Key preprocessing steps included lip segmentation, Gaussian blurring, contrast stretching, bilateral filtering, and sharpening. These techniques were pivotal in improving image quality, reducing noise, and emphasizing relevant features, thereby facilitating more accurate lip movement interpretation.

The dataset was divided into training and testing subsets, with an 80-20 split, and the model was trained on cloud-based platforms like Kaggle due to hardware limitations. The 3D CNN architecture consisted of multiple Conv3D layers with subsequent MaxPooling3D layers, followed by dense layers with dropout regularization to prevent overfitting.

The model demonstrated high accuracy in classifying lip movements, achieving a training accuracy of 97.4% and a testing accuracy of 99.2%. These results indicate that the proposed model is highly effective in recognizing spoken words from lip movements with minimal error. The study highlights the importance of spatiotemporal feature extraction in improving the accuracy of lip-reading systems and paves the way for further advancements in visual speech recognition.

2.2 Arabic Lip Reading with Limited Data Using Deep Learning

Arabic lip reading poses a unique challenge due to the scarcity of large-scale, labeled datasets in this language. To address this issue, the paper proposes an end-to-end system built for limited-data scenarios, focusing on Modern Classical Arabic (MCA). Unlike many existing lip-reading solutions in English—which rely heavily on massive datasets—this approach introduces a specialized visual model (a CNN trained on "viseme" images) that exploits repeated frames in short video clips. This CNN, once trained, is integrated with a temporal model (GRUs) to handle sequence dynamics in spoken words.

By gathering a new dataset of 20 Arabic words, each uttered by 40 native speakers, the authors demonstrate how repeated mouth shapes (visemes) can be systematically extracted to boost the amount of training material without substantially increasing recording efforts. This "viseme classifier" not only learns the subtle differences across Arabic phonemes (mapped to 10 viseme categories) but also serves as a pretrained front-end for downstream word-level recognition.

The authors begin by collecting and preprocessing data: they record continuous videos of volunteers uttering 20 MCA words (e.g., digits, weekdays, frequent daily words). Each one-second clip is converted to frames, and a facial landmark detection algorithm (dlib) identifies the mouth region. By cropping and resizing these images, they obtain a uniform Region of Interest (ROI) that captures lip motions from frame to frame.

In the next phase, they extract repeated visemes from each clip by identifying frames where mouth shapes remain relatively stable. Clustering such frames yields multiple instances of each lip shape, which are then grouped into 10 predefined viseme classes (spanning all 28 Arabic phonemes). A CNN architecture is then trained on this expanded set of labeled viseme images, effectively learning spatial patterns specific to Arabic lip positions and movements.

For full word recognition, the authors design a two-part end-to-end system. The previously trained CNN becomes a frozen front-end: its classification layer is removed, and intermediate "bottleneck" features are passed to a tem-

poral model. This second model uses a GRU-BiGRU arrangement, which integrates context across sequences of frames. To mitigate overfitting, data-augmentation (rotations, flips, brightness changes) is applied to each word clip, increasing the effective size of the training set. Finally, the combined system outputs the most likely Arabic word class.

The proposed system shows promising performance despite the limited dataset size. On the newly collected 20-word dataset, the CNN + GRU pipeline achieves an accuracy of 83.02%, successfully differentiating among words with similar phoneme sets (e.g., ثلاثة / ثلاثاء). Moreover, the pretrained viseme classifier generalizes well: when applied to an existing Arabic lip-reading dataset from Dweik et al., it yields an 85.81% accuracy—about 3% higher than the original method for that corpus.

Additional experiments reveal that the CNN alone can distinguish person identities (99.77% accuracy) based on individualized lip shapes, underscoring the richness of viseme data. Overall, the ability to train a specialized model from scratch, then reuse it for new word sets, suggests a practical blueprint for future Arabic lip-reading systems. The authors foresee extending this strategy to sentence-level recognition and collecting more diverse data, aiming to broaden the coverage of Arabic dialects and reduce dependence on large-scale external resources.

2.3 We Hear What They Say: Lip Reading in Arabic for the Voice Impaired System (LRAVI)

The Lip Reading in Arabic for the Voice Impaired System (LRAVI) is designed to enhance communication for individuals with hearing or speaking impairments by utilizing Visual Speech Recognition (VSR). Unlike many existing lip-reading systems that focus primarily on English and rely on extensive datasets, LRAVI targets the Arabic language, which presents unique challenges due to its linguistic complexity and the limited availability of large-scale labeled datasets. The system is capable of recognizing 100 common Arabic words by analyzing mouth movements from video inputs and converting these visual cues into text or audio outputs. This not only bridges the communication gap for the voice-impaired community but also improves speech recognition accuracy in noisy environments. The primary contributions of this work include the creation of a comprehensive Arabic lip-reading dataset, the development of advanced preprocessing techniques, the implementation of sophisticated deep learning models optimized for Arabic, and the integration of visual and backend models into an end-to-end training framework.

The implementation of LRAVI involves three main stages: dataset collection, preprocessing and splitting, and model training and integration. For dataset collection, two distinct datasets were compiled: an Arabic dataset comprising 24,700 videos from 86 native Arabic speakers (24 females and 62 males) sourced from YouTube and direct camera recordings, and an English dataset from BBC's LRS2 and LRS3 containing 164,000 videos to facilitate transfer learning. During preprocessing, videos were standardized to 1-2 seconds, converted into 30 grayscale frames, and processed to isolate the mouth region. Each frame was resized to 112x112 pixels and normalized. The dataset was then split into training (17,640 videos), validation (360 videos), and testing (710 videos) sets. The model architecture consists of a visual frontend employing 3D convolutions followed by a 2D ResNet to extract spatiotemporal features, and a backend transformer-based model with multi-head self-attention layers to interpret these features. Training involved pretraining the backend model on the English dataset, finetuning with the Arabic datasets, and integrating both frontend and backend models into an end-to-end framework. Additionally, a spell correction mechanism was implemented to enhance prediction accuracy by matching model outputs with the closest correct words from a predefined vocabulary.

The LRAVI system demonstrated competitive performance on the testing dataset, achieving a Word Error Rate (WER) of 39.8% and a Character Error Rate (CER) of 28.1% when evaluated on videos from unseen speakers. Precision and recall varied across the 100 Arabic words, with some words like "سيارة" (car) and "مستشفى" (hospital) attaining perfect recall and high precision, indicating the model's effectiveness in accurately recognizing frequently used terms. However, certain words exhibited lower precision and recall, highlighting areas for improvement in model accuracy and robustness.

3 Dataset

3.1 Data Collection

Our proprietary dataset focuses on ten Arabic emergency words, each spoken by eleven (11) individuals, with three (3) repetitions per individual. This yields a total of 330 video clips (10 words × 11 speakers × 3 repetitions). Each clip lasts about 1–2 seconds and is strictly visual (no audio) to capture only the speaker's lip movements.

Because this dataset is curated for emergency scenarios, it includes commonly used Arabic words such as "اسعاف", "حريق", and "النجدة". All clips were recorded under controlled conditions, ensuring consistent lighting and minimal background noise. However, the dataset is proprietary and thus not publicly available for direct download.

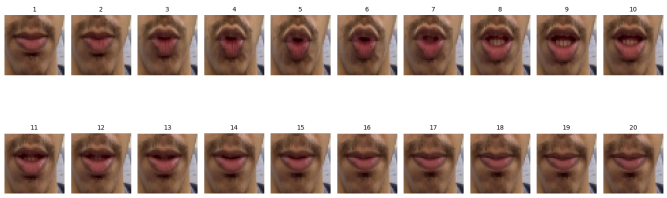

Figure 4: Illustrative example of SHARES's dataset, saying the word.شرطة

3.2 Data Cleaning

Before any analysis, we ensure:

1. File Format Consistency: All video files are stored in standard formats (e.g., .mp4, .avi).
2. Corrupted Files Check: Invalid or corrupted videos are logged and excluded.
3. Directory Structure: Each emergency word is in its own folder. Within each word folder, subfolders contain the raw videos (and later, extracted frames).
4. Naming Conventions: Folders and filenames follow a uniform scheme to simplify downstream processing.

3.3 Data Preparation

We process each video by extracting frames at fixed intervals, resizing them, and converting them into an easily accessible structure for deep learning pipelines.

- Frame Extraction: We fix the number of frames to 20 per video, capturing evenly spaced images from start to finish.
- Resizing: Every extracted frame is resized to 224×224 pixels.
- Normalization: Pixel values are scaled to [0,1].

3.4 CSV Mapping (DataFrame Creation)

To streamline the loading process, we generate a CSV file that maps each word label to the path where the frames are stored. This structure yields 330 rows × 2 columns (Word, Frames Directory), corresponding to:

$$10\,\text{words} \times 11\,\text{speakers} \times 3\,\text{clips per speaker} = 330\,\text{total}$$

3.5 Data Preprocessing and Splits

Once the frames for each video have been extracted and validated, we arrange them into NumPy arrays (X) and their corresponding label encoder (y). For each of the 330 video clips, exactly 20 frames of size 224×224 pixels are used, yielding a final dataset shape of (330,20,224,224,3) for the inputs and (330,) representing 10 classes corresponding to the 10 Arabic emergency words. Each output is encoded as an integer label (0 to 9). To evaluate the model fairly, we split these data into three sets:

- Training set (Train) for model fitting,
- Development set (Dev) for hyperparameter tuning and early stopping,
- Testing set (Test) for final performance assessment.

Dataset	# of Samples	Shape
Train	264	(264, 20, 224, 224, 3)
Dev	33	(33, 20, 224, 224, 3)
Test	33	(33, 20, 224, 224, 3)

Table 1: Statistics of the SHARES dataset.

4 Methodology

This section describes the design and implementation of SHARES. SHARES relies on modern deep learning to accurately interpret lip movements for emergency-related Arabic words. Given that traditional 2D convolutional models struggle with the temporal dimension inherent in videos, we chose to fine-tune a ResNet R(2+1)D architecture. This model naturally handles both spatial and temporal information by splitting 3D convolutions into separate 2D and 1D operations, making it more efficient and effective for video-based tasks compared to purely 3D CNNs.

4.1 Data Augmentation

To address the limited size of our dataset (330 video samples) and the subtle nature of lip movements, we implemented a tailored data augmentation pipeline to artificially increase sample variability. The aim is to encourage the model to learn robust, invariant features critical for recognizing Arabic emergency words, rather than overfitting to specific frames or lighting conditions.

Augmentation Strategy. Our custom `VideoAugment` class applies a series of spatial and color-based transformations to each video clip, either:

- *Independently on each frame,* creating high variability between consecutive frames, or
- *Consistently across all frames,* preserving the temporal coherence of natural head movements.

Specifically, we employ:

- **Random Rotations (up to ±2° or ±8°).** These mild rotations simulate slight head tilts without

warping the lip region excessively. When consistent across all frames, the spatial-temporal integrity of a lip movement sequence is maintained, closely mimicking natural speaking motions.

- **Random Horizontal Flips (with probability 0.2).** Reflecting frames helps account for variations in speaker orientation and camera angles.

- **Color Jitter.** Small random adjustments in brightness and contrast (*e.g.,* ±20%) address differing illumination conditions, ensuring the model focuses on lip shape and motion instead of overfitting to color intensity.

By introducing small but diverse variations in orientation and appearance, this augmentation pipeline helps the model generalize to new speakers, slightly different camera angles, and natural head movements. In particular, preserving temporal context (via consistent transformations) is crucial for a coherent lip-reading model, while frame-independent transformations can further enrich variability when limited training data is available.

4.2 Data Preprocessing

To feed our video clips into the deep learning pipeline, we structure them using a custom `LipReadingDataset` class. This class handles three primary tasks:

1. **Shape Rearrangement:** The raw input `X` of shape (N, T, H, W, C) is converted into the format (N, C, T, H, W), aligning with PyTorch's channel-first convention for video models.

2. **Label Management:** Each sample's label `y` is stored as a `long` integer, making it compatible with cross-entropy loss functions. (If the labels were one-hot encoded, a conversion back to integer indices would be performed here.)

3. **Optional On-the-Fly Transformations:** A user-specified `transform` can be applied to the clip. In our case, we incorporate data augmentation (via `VideoAugment`) only in the training set, while validation and test data remain unaltered.

4. **Batch Loading and Splits.** We instantiate separate `Dataset` objects for training, development, and testing. The `DataLoader` class then batches these samples (here, 8 videos per batch) and shuffles them during training to prevent overfitting and improve generalization. Meanwhile, the development and test sets are loaded in deterministic (unshuffled) order for consistent evaluation.

This design ensures a clean separation between training and evaluation data, while also providing a flexible mechanism to integrate data augmentation steps only when necessary. By handling shape formatting, label indexing, and transformations transparently, `LipReadingDataset` simplifies the preprocessing pipeline and allows for seamless experimentation with different augmentation strategies.

4.3 Model Architecture and Fine-Tuning

Our lip-reading model is based on the ResNet R(2+1)D architecture, an 18-layer network originally introduced to more effectively decouple spatial and temporal convolutions. Conceptually, each 3D convolution layer is factorized into two separate operations: one for spatial features and another for temporal features. The backbone design follows a structure akin to ResNet-18, but each residual block is adapted to operate on video clips rather than single images. Figure 2 conceptually illustrates how this factorization works:

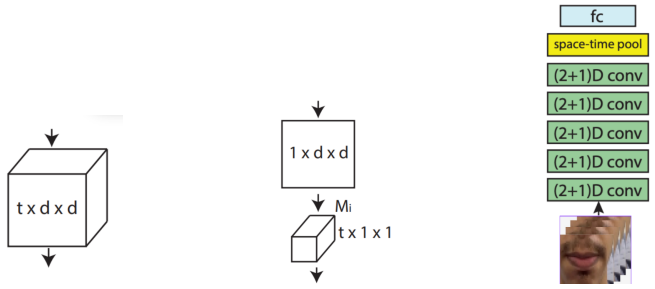

Figure 5: **(2+1)D vs 3D convolution** (a) Input Volume ($t \times d \times d$). (b) Factorized Convolution ($1 \times d \times d + t \times 1 \times 1$). (c) Complete R(2+1)D Network.

- **(a) Input Volume ($t \times d \times d$).** Full 3D convolution is carried out using a filter of size ($t \times d \times d$) where t denotes the temporal extent and d is the spatial width and height.

- **(b) Factorized Convolution ($1 \times d \times d + t \times 1 \times 1$).** R(2+1)D splits the 3D kernel into a 2D spatial kernel ($1 \times d \times d$) followed by a 1D temporal kernel ($t \times 1 \times 1$). This reduces the number of parameters and can improve training stability by separately learning spatial and temporal feature representations.

- **(c) Complete R(2+1)D Network.** After a series of such factorized layers, a global space–time pooling operation is performed before feeding the resultant feature map into a fully connected layer for classification. The backbone follows a ResNet-like structure but replaces each 3D block with two separate convolutions (2D + 1D).

Pretrained Weights and Transfer Learning. To leverage established spatiotemporal features, our model is initialized with weights from a large-scale video dataset (e.g., Kinetics). These pretrained weights give the network a strong baseline for recognizing temporal patterns, reducing the training burden on a smaller, domain-specific dataset such as our Arabic emergency lip-reading corpus.

Freezing and Fine-Tuning. We selectively freeze early layers to preserve generic spatiotemporal features and only fine-tune higher-level blocks (e.g., `layer4`) along with the final fully connected layers. This strategy mitigates overfitting and speeds up convergence when training data is limited. Specifically, we set `requires_grad = False` for most backbone parameters except those in `layer4` and the classification head.

Final Classification Layers. The original network's last fully connected layer is replaced with a lightweight head consisting of a linear layer (output dimension 32), batch normalization, ReLU, dropout (0.5), and a final linear layer projecting to our N classes (here, $N = 10$). This design captures sufficient complexity to learn the distinguishing lip-movement cues for Arabic emergency words while limiting excessive capacity that could cause overfitting.

```
import torch.nn as nn
from torchvision.models.video import r2plus1d_18, R2Plus1D_18_Weights

class LipReadingModel(nn.Module):
    def __init__(self, num_classes, freeze_backbone=True):
        super().__init__()
        self.base_model = r2plus1d_18(weights=R2Plus1D_18_Weights.DEFAULT)

        # Optionally freeze most layers to preserve pretrained spatiotemporal features
        if freeze_backbone:
            for name, param in self.base_model.named_parameters():
                if "layer4" not in name and "fc" not in name:
                    param.requires_grad = False

        # Replace the final classifier
        in_features = self.base_model.fc.in_features
        self.base_model.fc = nn.Sequential(
            nn.Linear(in_features, 32),
            nn.BatchNorm1d(32),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(32, num_classes)
        )

    def forward(self, x):
        return self.base_model(x)
```

Listing 1: Definition of the fine-tuned R(2+1)D model architecture.

By focusing on the last few layers and maintaining core spatiotemporal filters from pretrained R(2+1)D blocks, we can efficiently adapt to the unique patterns of Arabic lip movements. This approach yields improved generalization and faster convergence compared to training a 3D CNN from scratch.

4.4 Implementation Details

Our training pipeline is constructed in `PyTorch` and integrates several best-practice components to maximize efficiency and stability:

- **Loss Function:** We use `CrossEntropyLoss` to handle multi-class classification across the 10 Arabic emergency words.
- **Optimizer:** The `AdamW` algorithm (lr $= 1 \times 10^{-4}$, weight_decay $= 1 \times 10^{-4}$) is employed, balancing gradient updates with an effective form of L2 regularization.
- **Learning Rate Scheduling:** A `ReduceLROnPlateau` scheduler monitors the development-set loss and halves the learning rate if improvement stalls for two consecutive epochs.
- **Early Stopping:** If the development-set loss fails to improve for `patience` epochs (set to 30 in our experiments), training is halted to prevent overfitting.
- **Training Procedure:**
  - We train for up to 50 epochs, repeatedly iterating through mini-batches from the `train_loader`.
  - After each epoch, the model is evaluated on the development set (`dev_loader`), and key metrics (loss, accuracy) are tracked.
  - A checkpoint of the model is saved whenever its development-set accuracy exceeds the previous best.
- **Hardware:** Experiments are conducted on a GPU-enabled system to handle the computational demands of 3D-like convolutional operations.

4.5 Model Evaluation

After training, the model is evaluated on both the test and development sets using several performance metrics: accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of the model's effectiveness across various words.

Accuracy: This metric represents the ratio of correctly predicted observations to the total number of observations, giving an overall measure of the model's performance. It is defined as:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

For instance, in SHARES, if the task is to classify Arabic lip movements (e.g., حريق, مساعدة, إسعاف), a 90% accuracy indicates that 9 out of 10 samples are correctly identified.

Precision (Positive Predictive Value): Precision measures the accuracy of positive predictions. It is the ratio of true positive predictions to the total number of positive predictions made:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

A high precision in SHARES minimizes false alarms. For example, if the system detects "خطر," a high precision ensures that this prediction is likely to be correct.

Recall (Sensitivity, True Positive Rate): Recall quantifies the model's ability to identify all relevant instances. It is defined as the ratio of true positive predictions to the total actual positives:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

In the context of SHARES, a high recall means that when a speaker utters اهرب, the system is likely to detect and recognize this critical emergency word.

F1-Score (F1-Measure): The F1-Score is the harmonic mean of precision and recall. This balanced metric considers both false positives and false negatives, offering a single measure that reflects the overall performance:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

In SHARES, the F1-Score ensures a balance between avoiding false alarms and capturing genuine emergencies, providing a comprehensive overview of the system's performance.

In addition to these metrics, a confusion matrix is employed to visualize the performance across different classes. This visualization helps identify words that are frequently misclassified, offering critical insights for further refinement of the model.

4.6 Challenges and Mitigation Strategies

Several challenges arise in the development of effective lip-reading models, particularly with a limited dataset. To counter the risk of overfitting due to the small number of training samples, we implement extensive data augmentation and leverage transfer learning from a pre-trained model. Variability in speaker expressions and minor differences in video quality are addressed by collecting data under controlled conditions and employing robust augmentation techniques to ensure the model's generalizability. The high computational demands of processing video data and applying data augmentations are managed through GPU acceleration and parallel processing techniques. Furthermore, the R(2+1)D architecture's unique design, which separates spatial and temporal convolutions, proves essential for capturing the dynamic features critical for accurate lip-reading.

5 Result

After training the model for 50 epochs, we first analyze the training progress by examining the accuracy and loss over epochs. The combined plot in Figure 6 shows that while the training accuracy steadily increases, the validation accuracy levels off and the loss begins to diverge slightly. This behavior indicates a bit of overfitting.

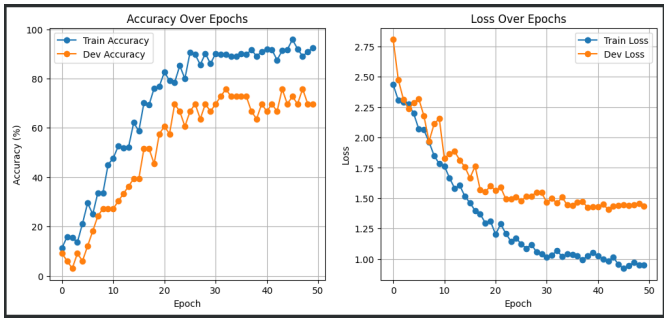

Figure 6: Training and Validation Accuracy and Loss over Epochs. Note the slight overfitting as the validation performance plateaus while training performance continues to improve.

5.1 Evaluation on the Development Set

Figure 7 presents the confusion matrix for the development set. This matrix visualizes the distribution of correct and incorrect predictions across different words, helping to identify areas where the model might be misclassifying similar lip movements.

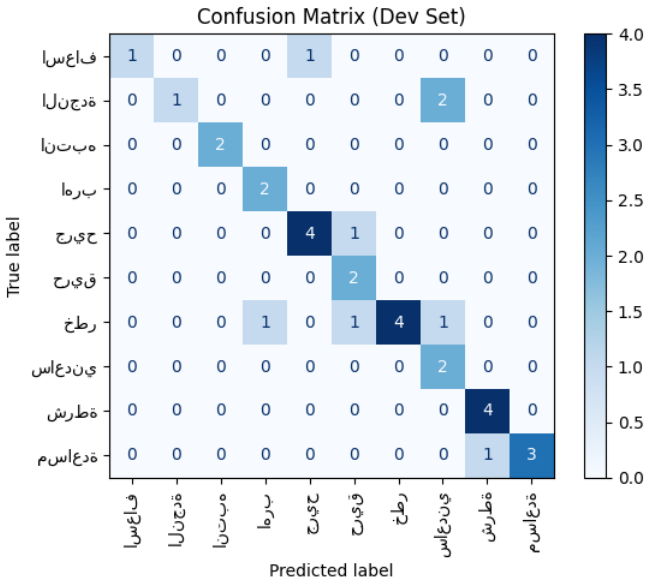

Figure 7: Confusion Matrix for the Development Set

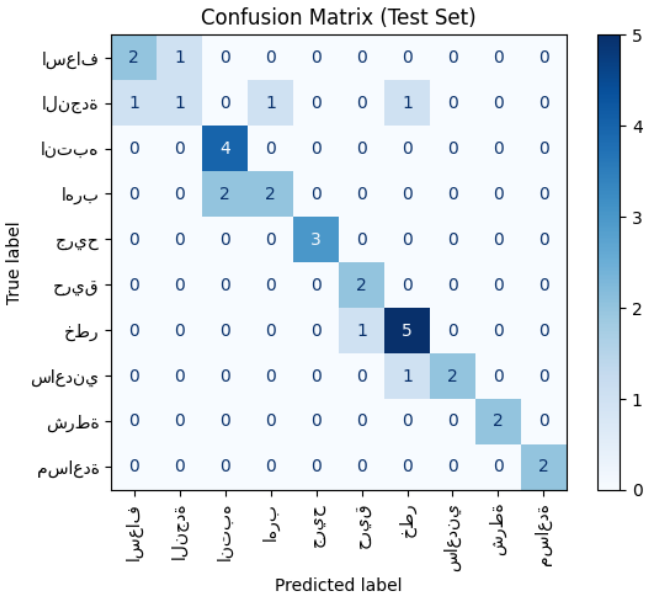

Figure 8: Confusion Matrix for the Test Set

Table 2: Classification Report for the Development Set

Class	Prec.	Rec.	F1	Supp
اسعاف	1.00	0.50	0.67	2
النجدة	1.00	0.33	0.50	3
انتبه	1.00	1.00	1.00	2
اهرب	0.67	1.00	0.80	2
جيح	0.80	0.80	0.80	5
حيق	0.50	1.00	0.67	2
خطر	1.00	0.57	0.73	7
ساعدني	0.40	1.00	0.57	2
شرطة	0.80	1.00	0.89	4
مساعدة	1.00	0.75	0.86	4
accuracy		0.76		
macro avg	0.82	0.80	0.75	33
weighted avg	0.86	0.76	0.76	33

Table 3: Classification Report for the Test Set

Class	Prec.	Rec.	F1	Supp.
اسعاف	0.67	0.67	0.67	3
النجدة	0.50	0.25	0.33	4
انتبه	0.67	1.00	0.80	4
اهرب	0.67	0.50	0.57	4
جيح	1.00	1.00	1.00	3
حيق	0.67	1.00	0.80	2
خطر	0.71	0.83	0.77	6
ساعدني	1.00	0.67	0.80	3
شرطة	1.00	1.00	1.00	2
مساعدة	1.00	1.00	1.00	2
accuracy		0.76		
macro avg	0.79	0.79	0.77	33
weighted avg	0.76	0.76	0.74	33

Next, we evaluate the model using confusion matrices on the test sets.

5.2 Evaluation on the Test Set

Figure 8 shows the confusion matrix for the Test set. This visualization highlights how well the model is classifying the different words and points out specific areas where misclassifications occur.

6 Discussion

The results obtained from SHARES indicate a promising direction for Arabic lip-reading in emergency scenarios. The system achieves a test accuracy of 76%, demonstrating its capability to recognize critical emergency words from lip movements. However, several insights and challenges emerge from the analysis of the results.

6.1 Analysis of Results

Examining the confusion matrices and classification reports, certain words exhibit higher misclassification rates. Words with similar lip movements, such as "نجدة" (help) and "مساعدة" (assistance), tend to be confused with each

other. This suggests that the model struggles to distinguish between phonemes with minimal visual distinction. Additionally, the presence of individual speaker variations affects performance, as different lip shapes and articulation styles can introduce variability in predictions.

The high recall scores for words such as "اسعاف" (ambulance) and "شرطة" (police) indicate that the model correctly identifies these words when spoken. However, lower precision in certain words implies that the model sometimes assigns incorrect labels to other classes. This suggests a need for more refined feature extraction and improved discriminative learning for visually similar words.

6.2 Implications of Findings

The implications of these findings extend to real-world applications where quick and accurate recognition of emergency words is crucial. The current model demonstrates feasibility but requires enhancements to ensure reliability in practical deployments. The observed overfitting during training indicates that additional regularization techniques, such as dropout tuning and more aggressive data augmentation, could improve generalization. Additionally, incorporating external datasets or transfer learning from larger pre-trained lip-reading models may help boost performance.

In scenarios where real-time lip-reading is essential, SHARES must optimize computational efficiency. Deploying lightweight versions of R(2+1)D CNNs or exploring transformer-based architectures such as Vision Transformers (ViTs) could improve real-time performance. Moreover, fusing audio signals (when available) with visual inputs may enhance accuracy by providing multimodal cues.

6.3 Potential Improvements

Several improvements can be made to enhance the SHARES system:

- **Expanding the dataset:** Increasing the number of speakers, covering diverse dialects and varied lighting conditions, will help improve robustness.
- **Advanced augmentation techniques:** Applying GAN-based synthetic data generation and lip-movement interpolation techniques can help create additional training samples.
- **Hybrid modeling approaches:** Integrating recurrent neural networks (RNNs) such as LSTMs or GRUs alongside CNNs may capture sequential dependencies better.

7 Limitations

Despite the promising performance of SHARES, several constraints limit its scalability and broader adoption in real-world settings:

- **Limited Dataset Size.** SHARES relies on a proprietary collection of only 330 video samples, restricting the model's ability to generalize to different speakers, accents, and ambient conditions. While data augmentation partially mitigates overfitting, substantially larger and more varied datasets are needed to capture the full range of Arabic phonetic and visual nuances.
- **Hardware and Cost Constraints.** Training and deploying R(2+1)D CNNs demands powerful GPU resources, which can be expensive and infrastructurally intensive. Such constraints may limit SHARES's feasibility in cost-sensitive or resource-constrained environments, like smaller clinics or rural areas.
- **Dialectical and Contextual Variability.** Arabic consists of multiple dialects and colloquialisms that differ from Modern Standard Arabic (MSA). SHARES currently focuses on MSA-based emergency vocabulary. Scaling to other dialects or larger vocabularies requires significant additional data collection and model retraining.
- **Real-World Environmental Factors.** Videos in this study were recorded under relatively controlled lighting and minimal occlusion. In practical scenarios, occlusions (e.g., face masks), poor illumination, and rapid head movements can degrade lip-reading accuracy, necessitating more robust image processing and domain adaptation strategies.

8 Conclusion

This paper introduced SHARES, an Arabic lip-reading emergency system that leverages deep learning techniques to recognize emergency words from lip movements. By utilizing the R(2+1)D CNN model, SHARES demonstrated an accuracy of 76% on a proprietary Arabic dataset, highlighting its potential in aiding individuals with speech and hearing impairments during emergencies.

Despite its promising performance, challenges such as misclassification of visually similar words, variability in speaker articulation, and the need for a larger dataset remain. Future work should focus on expanding the dataset, employing hybrid models, and refining feature extraction techniques to improve accuracy and robustness. Furthermore, integrating real-time deployment strategies and exploring multimodal fusion with audio cues could significantly enhance system effectiveness.

SHARES represents an important step in Arabic lip-reading research, contributing to the broader field of assistive technologies. With further advancements, it has

the potential to be deployed in real-world emergency response applications, bridging communication gaps for individuals in critical situations.

References

[1] A. Ye and S. Kwong, *"Computer Vision Lip Reading,"* Project Documentation, https://docs.google.com/document/d/1FLVwjXf4BfxgjIBl9CszCMwwwQ-Tm0crAv71qGPmLCM/edit?tab=t.0, last accessed on Aug. 20, 2023.

[2] Z. Jabr, S. Etemadi, and N. Mozayani, *"Arabic Lip Reading with Limited Data Using Deep Learning,"* *IEEE Access*, vol. XX, no. XX, pp. 1–16, 2017, doi: 10.1109/ACCESS.2022.DoiNumber.

[3] M. A. Shaker, M. Gasser, M. Elsaid Moussa, M. Ashraf, O. Mokhtar, M. Kotb, and M. Mamdouh, *"We Hear What They Say: Lip Reading in Arabic for the Voice Impaired System (LRAVI)"*, in *Proceedings of [Conference Name]*, Cairo, Egypt, Oct. 2023, pp. 1–8. [Online]. Available: https://www.researchgate.net/publication/374752831

[4] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, *"A Closer Look at Spatiotemporal Convolutions for Action Recognition,"* in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 6450–6459.

[5] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, *"MediaPipe: A Framework for Building Perception Pipelines,"* *arXiv preprint arXiv:1906.08172*, 2019. [Online]. Available: https://arxiv.org/abs/1906.08172

[6] J. S. Chung, A. W. Senior, O. Vinyals, and A. Zisserman, *"Lip Reading Sentences in the Wild"*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp. 3444–3453.

[7] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, *"Deep Audio-Visual Speech Recognition"*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 195–213, 2019.

[8] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, *"Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-Offs in Video Classification"*, in *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, 2018, pp. 305–321.

[9] D. Tran, J. Ray, Z. Shou, S. Chang, and M. Paluri, *"ConvNet Architecture Search for Spatiotemporal Feature Learning"*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 10, pp. 2429–2443, 2019.

[10] J. Carreira and A. Zisserman, *"Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset"*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp. 6299–6308.

[11] I. A. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, *"LipNet: End-to-End Sentence-level Lipreading"*, presented at the *GPU Technology Conference (GTC)*, San Jose, CA, 2017. [Online]. Available: https://arxiv.org/abs/1611.01599

[12] M. A. Masaeed and S. Al-Qatawneh, *"Automatic Arabic Lip Reading System Based on Deep Learning"*, *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 8, no. 1, pp. 1–14, 2022.

[13] J. Carreira and A. Zisserman, *"Quo vadis, action recognition? A new model and the Kinetics dataset"*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp. 6299–6308.

[14] S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[15] Z. Qiu, T. Yao, and T. Mei, "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 5533–5541.

[16] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, pp. 4489–4497.