Deaf and dumb Muslims cannot reach advanced levels of education due to the impact of obstruction on their educational attainment. This leads to their inability to learn, recite, and understand the meanings and interpretations of the Holy Qur’an as easily as ordinary people, which also prevents them from applying Islamic rituals such as prayer that require learning and reading the Holy Qur’an. In this paper, we propose a new model for Qur’anic sign language recognition based on convolutional neural networks through data preparation, preprocessing, feature extraction, and classification stages. The proposed model is aimed at recognizing the movements of the Arabic sign language by recognizing the hand gestures that refer to the dashed Qur’anic letters in order to help the deaf and dumb learn their Islamic rituals. The experiments have been conducted on a part of a large Arabic sign language dataset called ArSL2018, which represents the 14 dashed letters in the Holy Qur’an, so that this part contains only 24,137 images. The experimental results demonstrate that the proposed model performs better than the other existing models.

1. Introduction

Deaf and dumb people use sign language to communicate with others in their daily lives. Sign language is uncommon outside the deaf community, and communication between deaf and hearing people is a major challenge. Some hearing parents have deaf children, which generates a language gap. Deaf children are difficult to raise, nurture, and teach Islamic customs [1]. Numerous Arabic sign languages (ArSLs) use the same alphabetic. Deaf and dumb Arabs use Egyptian, Jordanian, Tunisian, and Gulf sign languages. Lack of information, an inability to communicate, and an inability to perform religious ceremonies create a gap. All of ArSL’s issues need machine translation to enable the deaf to attend school and get scientific knowledge in their native language [2, 3]. Pattern recognition in human-computer interaction shifts to computer vision and machine learning. Therefore, deaf hand gestures are a major tool for recognizing Qur’anic alphabetical letters.

In this paper, we focus on Qur’anic sign language recognition systems (QSLRS), which enable the deaf and dumb community to overcome communication challenges, learn the Islamic rituals, and learn the Arabic alphabet, which is the language of the Holy Qur’an. Through the use of QSLRS,
they can identify the 29 Qur’anic surahs with the dashed letters, which are 14 letters arranged in alphabetical order: ﷲ ﷩ ﷪ ﷯ ﷪ ﷯ ﷮ ﷭ ﷮ ﷯ ﷰ ﷱ ﷲ ﷳ. However, according to previous studies, this remains a challenge for academics and researchers due to the fact that the automated detection algorithms for ArSLs are inaccurate and the recognition field is small. Therefore, we propose a new model for QSLRS based on convolutional neural networks (CNNs) through

![Image](image_url)
data preparation, preprocessing, feature extraction, and classification stages in order to help the deaf and dumb learn their Islamic rituals, overcome challenges of communication with others, and learn the Arabic alphabetical characters. The main contributions of this paper are as follows:

(i) Identification of fixed alphabetic signs for ArSL in order to help the deaf and dumb learn the Qur’anic surahs that begin with dashed letters

(ii) Formation and construction of a new Qur’anic sign language (QSL) dataset built from a large ArSL dataset called ArSL2018, which consists of only 24137 samples representing 14 dashed letters from the beginnings of the Holy Qur’an surahs to implement the proposed model

(iii) The use of image augmentation approaches makes the proposed QSLRS-CNN model work better in real-life scenarios and reduces overfitting

(iv) Previous research findings indicate that there are currently no studies published or available online on the subject of Qur’anic sign language recognition (QSLR). As a result, one of the purposes of this study is to advance the studies that are being conducted in this field and to stimulate the production of materials that can be used for further research in this field

The remainder of this paper is structured as follows: The related works are presented in Section 2. The proposed methodology is presented in Section 3. In Section 4, the experimental results are presented and the main findings are discussed. Lastly, the conclusion is outlined in Section 5.

2. Related Works

The literature has several strategies for human-computer sign language recognition. These devices increase communication by understanding gestures and signs. These techniques involve capture, preprocessing, gesture representation, feature extraction, and classification. This section explores movement recognition strategies in sign languages.

There are several research initiatives to create sign language recognition systems across the globe, including Arab countries, which rely on vision or sensor gloves. This work focuses on vision-based systems that recognize ArSL alphabets.

This section examines studies on the identification of the ArSL alphabet as well as the size of the datasets used in these studies. The algorithms or methodologies used by researchers are also discussed.

In [4], Ahmed et al. propose an ArSL translation system. As for the Arabic text system (ATASAT), this system relies on building two datasets for Arabic alphabet gestures. They introduce a new manual detection technique that detects and extracts Arabic sign gestures from an image or video depending on the hand’s coverage. They also use different statistical classifiers and compare the results to get a better classification.

In [1], they propose a machine learning-based Arabic sign language alphabet recognition system. They evaluate
2,800 images and 28 alphabets, with 10 participants in each class. There are 100 images for each letter, for a total of $28 \times 100 = 2,800$ images. Feature extraction is performed using a hand shape-based description, where each hand image is characterized by a vector of 15 values indicating key point locations, while classification is performed with K-nearest neighbors (KNN) and multilayer perceptron (MLP) algorithms. Testing shows 97.548% accuracy.

Luqman and Mahmoud [5] study Fourier, Hartley, and Log-Gabor transforms for ArSL recognition (ArSLR). The Hartley transform detects ArSLR with 98.8% accuracy using the support vector machine (SVM) classifier. Alzohairi et al. [6] can automatically recognize 63.5% of the movements of the Arabic alphabet by using a method that is based on images.

In 2020, Kamruzzaman [7] develops a vision-based approach for identifying Arabic hand sign-based characters and converting them into Arabic speech with a 90% recognition rate using CNN. ElBadawy et al. [8] propose a CNN-
based framework for recognizing 25 ArSL signs. On the basis of the training and testing data, this model has accuracy scores of 85 and 98 percent, respectively. Mohamed describes in [9] a computerized system that employs depth-measuring cameras and computer vision techniques to capture and segment images of facial expressions and hand gestures with a 90% recognition rate. Latif et al. [10] propose a computerized system that employs depth-measuring cameras and computer vision techniques to capture and segment images of facial expressions and hand gestures with a 90% recognition rate. Latif et al. [10] propose different CNN architectures using 54,049 sign images [11]. Their findings demonstrate the considerable influence that the size of the dataset has on the correctness of the model that is presented. The proposed model’s testing accuracy increases from 80.3 percent to 93.9 percent when the amount of the dataset is raised from 8,302 samples to 27,985 samples. A further improvement in the proposed model’s testing accuracy occurs when the amount of the dataset is increased from 33406 samples to 50,000 samples, resulting in a corresponding rise from 94.1 percent to 95.9 percent.

In [12], Alani and Cosma develop an Arabic sign recognition system based on the ArSL2018 dataset and a unique ArSL-CNN architecture. The accuracy of the suggested ArSL-CNN model during training is originally 98.80 percent, whereas the accuracy during testing is initially 96.59 percent. They decide to use a variety of resampling strategies on the dataset in order to mitigate the effect that imbalanced data has on the precision of the model. Based on the findings, the synthetic minority oversampling method (SMOTE) results in an improvement in overall testing accuracy from 96.59 percent to 97.29 percent.

Saleh and Issa [13] utilize the ArSL2018 to improve the accuracy of recognizing 32 hand motions from the ArSL-CNN using transfer learning and fine-tuning deep CNNs. To address the imbalance produced by class size disparity, the dataset is subjected to random undersampling. The total number of images is reduced from 54,049 to 25,600. The generated model has a 99.4 percent validation accuracy for the visual geometry group (VGG-16) and a 99.6 percent validation accuracy for the ResNet-152.

Shahin and Almotairi [14] suggest a deep transfer learning-based strong identification approach for ArSL. To reduce overfitting and improve performance, they use transfer learning techniques based on fine-tuning and data augmentation. The proposed residual network ResNet101 system achieves maximum accuracy with a score of 99.52 percent.

Abeje et al. [15] offer a unique sign language recognition system that converts Ethiopian sign language (ETHSL) to Amharic alphabets using computer vision and a deep CNN. The system receives sign language graphics and outputs Amharic. Preprocessing, feature extraction, and recognition make up the suggested system. The methodology includes data gathering, preprocessing, background normalization, picture scaling, ROI identification, noise reduction, brightness correction, and feature extraction. A deep CNN is utilized for end-to-end classification. The JPEG images were gathered under controlled conditions. Adjusting the image size and color reduced the running time. In addition, the findings reveal better recognition accuracy. The model achieves 98.5% training, 95.59% validation, and 98.3% testing accuracy.

Tamiru et al. [16] discuss the construction of an autonomous Amharic sign language translator utilizing digital image processing and machine learning methods. Preprocessing, segmentation, feature extraction, and classification are the four key system steps. Thirty-four characteristics are retrieved from the form, motion, and color of hand motions to depict Amharic sign symbols. Artificial neural networks (ANN) and multiclass SVM classification models are used. The recognition system can recognize Amharic alphabet signs with an average accuracy of 80.82 and 98.06 using ANN and SVM classifiers, respectively.

Despite recent advances in deep learning and the high precision of image categorization and prediction obtained with CNN, imbalanced data can have an impact on prediction model performance. Imbalanced data can have an

<table>
<thead>
<tr>
<th>Class name</th>
<th>Number of samples in the test data</th>
<th>Number of samples correctly classified</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alif</td>
<td>327</td>
<td>323</td>
<td>98.78%</td>
</tr>
<tr>
<td>Ha</td>
<td>312</td>
<td>307</td>
<td>98.40%</td>
</tr>
<tr>
<td>Ra</td>
<td>352</td>
<td>352</td>
<td>100.00%</td>
</tr>
<tr>
<td>Sin</td>
<td>323</td>
<td>315</td>
<td>97.52%</td>
</tr>
<tr>
<td>Sad</td>
<td>364</td>
<td>353</td>
<td>96.98%</td>
</tr>
<tr>
<td>Tah</td>
<td>379</td>
<td>370</td>
<td>97.63%</td>
</tr>
<tr>
<td>Ayn</td>
<td>427</td>
<td>420</td>
<td>98.36%</td>
</tr>
<tr>
<td>Qaf</td>
<td>348</td>
<td>331</td>
<td>95.11%</td>
</tr>
<tr>
<td>Kaf</td>
<td>341</td>
<td>320</td>
<td>93.84%</td>
</tr>
<tr>
<td>Lam</td>
<td>366</td>
<td>355</td>
<td>96.99%</td>
</tr>
<tr>
<td>Mim</td>
<td>368</td>
<td>353</td>
<td>95.92%</td>
</tr>
<tr>
<td>Nun</td>
<td>369</td>
<td>357</td>
<td>96.75%</td>
</tr>
<tr>
<td>Haa</td>
<td>304</td>
<td>295</td>
<td>97.04%</td>
</tr>
<tr>
<td>Ya</td>
<td>326</td>
<td>323</td>
<td>99.08%</td>
</tr>
</tbody>
</table>

Table 6: QSLRS-CNN accuracy on the test original data before applying sampling for 200 epochs.
influence on a model’s capacity to learn and its ability to be used in real-time scenarios. It is also worth looking at how sign language movements are translated into other mediums, such as writing and voice.

In the most current literature reviews, there are various research publications pertaining to ArSLR. In Table 1, we provide a concise description of the ArSLR systems that have been used in the past. In [2], [4], [6], [8], [10], [12–14], and [17–19], some proposed approaches and models for detecting the Arabic script have inadequate datasets. Latif et al. [11] presented the ArSL2018 dataset. This amounts to 54,049 samples.

3. Materials and Methods

In this section, we present a broad explanation of the architecture of the QSLRS-CNN, which is created to categorize the motions used in the QSL. In addition, it provides a description of the QSL dataset as well as the preprocessing methods that are used on the dataset.
3.1. QSL Dataset. In this study, a portion of a dataset called ArSL2018 [11] is used, which contains 24,137 images of the ArSL alphabet created by more than 40 people for 14 letters representing the openings of the Qur’anic surahs, which are letters arranged alphabetically (Alif، Ha ح، Ra ر، Sin س، Sad ص، Tah الط، Ayn ع، Qaf ق، Kaf ك، Lam ل، Mim م، Non ن، Haa ـه، and Ya ي). Before being employed by the proposed model, the images are preprocessed, and the dataset is separated into three groups for training, testing, and validation. It is then made available to researchers in machine learning and deep learning, consisting of grayscale images with dimensions of 64*64. It is executed in various forms of images with specific lighting and context, as illustrated in Figure 1. As mentioned earlier, it consists of a total of 14 output classes, ranging from 0 to 13, each representing an ArSL gesture. Table 2 shows the various classes along with their labels and number of samples available in the ArSL dataset for selected gestures.

In this study, we also use a part of the other ArSL dataset to test the proposed QSLRS-CNN model that is collected previously [4]. This dataset consists of 350 color images representing the gestures of 14 Arabic letters, with an average of 25 images per character gesture. As mentioned in [20], the dataset is taken in different ways, under different lighting conditions, and based on different signers with different hand sizes and wearing dark-colored gloves, as shown in Figure 2. Table 3 shows the various classes along with their labels and number of samples available in the ArSL dataset for the selected gestures.

In order to solve the undersampling problem when the dataset consists of classes of different sizes in terms of data elements, the class imbalance problem occurs, which leads to bias towards the majority class and negatively affects classification accuracy. To rectify the imbalance and eliminate bias, sample number fixing is used. ArSL2018 data samples are shown in Table 2. Ha ح contains 1526 images, whereas Ain ع contains just 2114. Before using QSLRS-CNN data, the number of samples is fixed.

In the proposed model, we used the ArSL2018 dataset along with the ArSL dataset, so we used the ArSL2018 dataset in the training phase only, and we combined the two datasets in the validation and testing phase, and we did not use the ArSL dataset in the training phase due to the lack of models in it. There are some experiments using the ArSL dataset in the training phase, which did not achieve the required or expected accuracy. So we preferred to use the ArSL2018 dataset in the training phase.

3.2. Dataset Problems. In machine learning, classification involves training a system using labeled datasets to classify an unknown dataset. In recent years, data has grown, but labeled data is scarce. Oversampling and undersampling are methods for changing a dataset’s class distribution (ratio of classes or categories). Most shallow machine learning
approaches rely on target classes having the same number of training examples. But in many cases, this assumption is wrong. The models favor the majority class and exclude the minority class since almost all instances are identified with one class and few with the other. When datasets are imbalanced, model performance suffers, and a class imbalance exists. In this circumstance, we may have good accuracy but poor precision, recall, and $F_1$-score [21].

Our dataset has an in-class imbalance. To balance the dataset, resampling is used. Resampling may undersample or oversample the dataset. Undersampling reduces the number of majority target samples. Oversampling involves developing new examples or repeating current ones while boosting minority class samples. Borderline SMOTE [22] is an example of an oversampling approach. In this paper, the unbalanced dataset of ArSL2018 is utilized with multiple machine learning models to explore oversampling and undersampling strategies and compare different evaluation measures. The next parts of this paper offer oversampling and undersampling findings for our proposed machine learning classification models.

3.3. Data Preparation and Preprocessing. Before testing the dataset, all gesture images are converted to 64*64 grayscale images. This removes RGB gesture image overprocessing. The image format has been changed from int8 to float32 for efficiency and speed of training, while the images may lose some information that can be retrieved [23, 24]. Images are standardized using 0–1 pixel values. The dataset is adjusted to meet CNN’s formatting requirements. The gesture images are randomly selected. The dataset is divided into a testing set (20%) and a training set (80%).

![Confusion matrix](image.png)

**Figure 9:** Confusion matrix of the proposed QSLRS-CNN model with SMOTE using 200 epochs.

![Model accuracy](image.png)

**Figure 10:** Accuracy of the proposed QSLRS-CNN model with SMOTE using 200 epochs.
3.4. Proposed QSL-CNN Model Architecture. The significance of deep learning and machine learning is rising rapidly in today’s world. The dataset is gathered from numerous sources throughout the process of creating the analytical model utilizing deep learning or machine learning. However, the data gathered cannot be utilized immediately to conduct the necessary analysis. In order to maximize the benefits of the machine learning and deep learning models, we need to make sure the data is in the appropriate format. Therefore, data preparation and preprocessing are done to solve this problem.

The general framework of the proposed CNN-based model consists of data preparation, data preprocessing, feature extraction, and classification stages as shown in Figure 3. The first stage is to identify dynamic and static gestures. Since deep learning models require data for training, gathering images to form a viable training set is the first step through data preparation stage. Data preprocessing is the next stage to perform transformations on each image for further diversification of each dataset. The final stage is to choose CNN-based deep learning model for feature extraction and classification on training and testing datasets.

CNNs, a deep learning approach, are good at image classification. Convolutional, pooling, and fully connected layers are utilized to develop CNN model architectures. Several of these layers are stacked to form a CNN.

We conclude the QSLRS-CNN proposed model with the output layer, which presents the ultimate classification result. This layer is comprised of 14 neurons, one for each of the 14 classes, and uses a softmax activation function. The graphical depiction of the proposed model can be seen in Figure 4, and Table 4 provides information about the model’s parameters.
4. Results and Discussion

The experiments use Keras packages and Python TensorFlow. QSLRS-CNN is trained on a system with an NVIDIA K80 GPU, 12 GB of RAM, and a 100 GB SSD. To eliminate bias, the training dataset is scrambled before being given to the network. The proposed QSLRS-CNN model is trained and tested using 14 classes from the original ArSL2018 dataset; the model is then trained and tested using alternative resampling strategies to solve class imbalance. Accuracy is used to evaluate the QSLRS-CNN technique. A signifies accuracy; TC and FC represent properly and wrongly categorized cases, respectively. Multiplying the computed amount by 100 gives a percentage as shown in

\[ A = \frac{TC}{TC + FC} \times 100. \]  

For a class, the accuracy can be determined using

\[ Ac = \frac{TCc}{TCc + FCc} \times 100. \]

TCc is the number of properly categorized examples from class c, while FCc is the number of wrongly classified instances. The final number is multiplied by 100 to determine each class’ accuracy.

4.1. QSLRS-CNN Performance Evaluation. Table 5 shows the performance of the proposed QSLRS-CNN model on 14 ArSL2018 classes for 100 and 200 epochs. The training dataset includes 24,137 images in 14 ArSL gesture groupings. Each training batch includes 128 samples. Each of the input and output layers contains 4,096 neurons. QSLRS-CNN is trained across numerous epochs. QSLRS-CNN obtains 97.13% accuracy after 100 learning epochs. In addition, QSLRS-CNN accuracy on the test original data before applying sampling for 200 epochs is shown in Table 6. Figure 5 shows the model’s accuracy after 100 epochs of training. Training and testing performances are similar across epochs, indicating no overfitting.

Figure 6 shows the QSLRS-CNN model’s training and testing accuracy curves after 100 epochs. Different epochs have similar training and testing performances. Also, Figure 6 shows QSLRS-100-epoch CNN’s confusion matrix. Off-diagonal entries in the confusion matrix represent mislabeled images. The sum of the confusion matrix’s diagonal values indicates classification accuracy.

4.2. QSLRS-CNN Performance Evaluation While Oversampling and Undersampling. In the previous section, QSLRS-CNN model findings are produced without data sampling. This study employs resampling approaches (oversampling and undersampling) to eliminate bias and address data imbalance in class distribution, which involves modifying the previous distribution for the minority and majority classes.

4.2.1. Oversampling. Oversampling generates synthetic samples from minority samples to correct class imbalances. This improves classification performance by increasing the quantity of minority class samples. Increasing minority class samples lengthens training. Random minority oversampling (RMO) randomly repeats samples from minority classes. The second is the synthetic minority oversampling method (SMOTE), which solves class imbalance by interpolating neighboring data points. Table 7 shows the QSLRS-CNN model’s findings on the ArSL2018 dataset after RMO and SMOTE. Applying oversampling methods boosts the QSLRS-CNN model’s efficiency. Using RMO, the proposed model achieves 98.37% training accuracy and 97.36% testing accuracy. Using SMOTE oversampling, the proposed model achieves 98.31% training accuracy and 97.67% testing accuracy.

Figure 7 shows model accuracy and training loss while using the SMOTE oversampling strategy. This graph demonstrates that the training and testing performances are close over various training and testing epochs, which indicates that the QSLRS-CNN model is not overfitting the data.

Figure 8 shows the confusion matrices of a 100-epoch QSLRS-CNN model trained using SMOTE oversampling. Overall, classification performance is good.

The confusion matrix that is produced by training the suggested QSLRS-CNN model using SMOTE for 200 epochs is shown in Figure 9. In the confusion matrix, the components that are diagonal indicate the number of properly classified images, while the ones that are off-diagonal signify the images that have been incorrectly labeled. The sum of the values on the diagonals of the confusion matrix is directly correlated to the degree of accuracy achieved by the classification.
Figure 10 shows how accurate the suggested QSLRS-CNN model is throughout different learning epochs following the use of SMOTE. According to the findings, the level of the model’s accuracy on both the training and the testing sets improves across all of the learning epochs, while training error rate of the proposed QSLRS-CNN model with SMOTE using 200 epochs is shown in Figure 11.

In addition, Tables 6 and 8 present the accuracy rating for each class. According to the findings of the experiments, QSLRS-CNN is able to achieve a higher level of classification effectiveness when the RMU, RMO, and SMOTE resampling methods are used. As an example, there are a total of 348 testing samples includes in the “Qaf” class prior to the use of the SMOTE resampling technique, and the accuracy is found to be 95.11%. Nevertheless, after implementing the
SMOTE resampling approach, the number of samples increases from 348 to 443 testing samples, which ultimately results to an improvement in accuracy from 95.11% to 97.52%. These findings validate the strong effect of using the SMOTE resampling approach as a solution to the imbalance issue and an improvement to the proposed model’s accuracy in general.

4.2.2. Undersampling. Undersampling of minorities at random is the second strategy for changing the distribution of samples across all of the classes in the ArSL2018 dataset with random minority undersampling (RMU). The dataset is not balanced until RMU excludes samples at random from classes with the majority membership. However, this may result in the loss of valuable information if there are fewer samples taken from members of the minority classes. Table 9 summarizes the findings that are obtained via the application of the QSLRS-CNN model to the ArSL2018 dataset after RMU. Through the use of the undersampling strategy, the proposed model is able to attain training and testing accuracies of 98.66 percent and 97.52 percent, respectively.

When the RMU approach is used, the model accuracy and training loss are shown in Figure 12.

Figure 13 displays the confusion matrices of the QSLRS-CNN model, which is trained with a total of 100 epochs and makes use of the RMU oversampling strategy.

4.3. Comparison with Other Models. The accuracy comparison of the proposed model with current state-of-the-art methods on the ArSL2018 dataset is provided in Table 10. The results demonstrate that the proposed QSLRS-CNN model outperforms current state-of-the-art methods in terms of accuracy when performing RMO, SMOTE, and RMU resampling on the dataset. The original CNN obtains an accuracy of 98.05% and 97.13% for training and testing, respectively. Latif et al. [10] obtain a training and testing accuracies of 97.6% and 97.1%, respectively, while Alani and Cosma [12] perform CNN to get training and testing accuracies of 98.80% and 97.29%, respectively. In comparison, the proposed model achieves a superior level of accuracy, highlighting the relevance of providing an appropriate amount of samples to improve the generalization efficacy of CNN while training deep learning models.

4.4. Analysis and Recommendations. This study develops a unique QSLRS-CNN architectural framework for Arabic sign identification. Experiments use the ArSL2018 dataset. 24137 pictures from 40 users were grouped into 14 categories. The QSLRS-CNN model has 98.05% training accuracy and 97.13% testing accuracy. The findings show the challenges of working with unbalanced data and the requirement to provide enough samples from each class to test and train deep learning models.

Unbalanced data also affects model accuracy. The dataset is tested using several resampling procedures. SMOTE increases test accuracy from 97.13% to 97.67%, which is statistically significant. The QSL-CNN model can be trained on a variety of ArSL to help Arabic-speaking deaf persons communicate. The results support the SMOTE oversampling approach for the ArSL2018 dataset. Our research is the first to employ SMOTE oversampling to assess class imbalance in the ArSL2018 dataset, which focuses on 14 letters representing Qur’anic surah openings. RMU gives QSLRS-CNN 98.66% accuracy. The proposed model’s 200-epoch results are accurate but take longer. The proposed model only uses fixed gestures to depict the discontinuous letters at the beginning of Qur’anic surahs. Isolated Qur’anic words or texts are also ignored.

We will test the QSLRS-CNN on various datasets and evaluate recurrent neural networks (RNN) and long short-term memory (LSTM) for the job in our future work. Arabic-speaking countries have many ArSLs that share alphabets. ArSL discrepancies may hinder communication. Thus, the future study will use transfer learning to build an improved ArSL deep learning model that works with ArSL variants. This method may help ArSL speakers overcome their challenges. We categorize the QSL alphabet letters that symbolize the first surahs of the Qur’an to make ARSL communication easier. To make the Holy Qur’an accessible to deaf people, a deep learning model should be created to translate its meanings into sign language.

5. Conclusions

The Arabic sign recognition system leverages part of the ArSL2018 dataset to propose the QSLRS-CNN model. Our collection includes 24,137 ArSL alphabet images from over 40 people, including 14 letters representing Qur’anic surah openings. QSLRS-CNN has 98.05 percent training accuracy and 97.13 percent testing accuracy. Unbalanced data and the need for sufficient class samples to train and evaluate deep learning models are highlighted. SMOTE obtains 97.67% accuracy on the ArSL2018 dataset, whereas RMU reaches 98.66%. The proposed model’s 200-epoch results are accurate but take longer. The proposed technology recognizes QSL movements by identifying hand gestures that refer to dashed Qur’anic letters to enable the deaf and dumb understand Islamic ceremonies. The experimental results show that the suggested model outperforms alternative models.

Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding author (Abdelmoty M. Ahmed, email: abd2005moty@yahoo.com) on reasonable request.

Conflicts of Interest

No potential conflict of interest was reported by the authors.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through a large group research project under grant number RGP2/246/44.
References


