

Quaternion tensor regression and applications

by

Adham Sami

A Thesis submitted to the Faculty of Graduate and Postdoctoral Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Statistics

University of Manitoba

Winnipeg

Copyright © 2026 by Adham Sami

Abstract

Most existing methods for video emotion recognition rely on deep learning models that require large labelled datasets and substantial computational resources. In small and medium scale settings, lightweight classification methods are of interest. Linear regression classifiers provide such an alternative, and recent work has shown that quaternion formulations can improve colour image classification by encoding Red, Green, and Blue (RGB) channels within a single algebraic entity. However, these methods have largely been restricted to static images represented as quaternion-valued matrices. In this thesis, we extend quaternion linear regression and collaborative representation classifiers to video emotion recognition by representing each clip as a third-order quaternion-valued tensor. We also develop a reduced biquaternion formulation with desirable algebraic properties and improved computational efficiency. Experiments on the Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D) and Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) benchmarks, including actor-independent and cross-dataset settings, are provided to evaluate the reduced biquaternion and quaternion linear regression classifier and reduced biquaternion and quaternion collaborative representation classifier models.

Acknowledgements

I would like to express my sincere gratitude to my advisors, Dr. Po Yang and Dr. Yang Zhang, for their guidance, support, and encouragement throughout this thesis. Their insightful feedback, patience, and expertise were instrumental in shaping this research and in helping me navigate the many challenges encountered along the way.

I would like to thank the committee members, Dr. Saumen Mandal and Dr. Hina Shaheen, for their time reviewing my paper, and for chairing my defence.

I would also like to thank the Department of Statistics at the University of Manitoba for providing an excellent academic environment and the resources necessary to complete this work. I am especially grateful to the faculty members who taught and prepared me throughout my studies, including Dr. Saman Muthukumarana, Dr. Mohammad Jafari Jozani, Dr. Saumen Mandal, Dr. Yuliya Martsynyuk, and Dr. Liqun Wang.

Finally, sincere gratitude also goes to the department staff members, Ms. Margaret Smith, Ms. Rosa Di Noto, and Ms. Sunny Joo, for their continuous support throughout my studies at the University of Manitoba.

Contents

Abstract	i
Acknowledgements	ii
List of Tables	v
List of Figures	vi
1 Literature Review	1
2 Preliminaries	6
2.1 Classification	6
2.2 Ridge Regression	7
2.3 Linear Regression vs. Collaborative Representation	7
2.4 Tensors	8
2.5 Tensor Regression	9
2.6 Quaternions	10
2.7 Reduced Biquaternions	11
2.8 Reduced Biquaternions vs. Hamilton Quaternions	15
3 Main Results	16
3.1 Quaternion Tensor Representation and Vectorization	17
3.2 Problem Setup and Data Representation	20

3.3	Real-Valued Reformulation	22
3.4	Quaternion Collaborative Representation (QCR)	23
3.5	Quaternion Linear Regression (QLR)	29
3.6	Reduced Biquaternion Collaborative Representation (RB-QCR)	32
3.7	Reduced Biquaternion Linear Regression (RB-QLR)	39
3.8	Classification via Collaborative Residuals	42
4	Experimental Results	44
4.1	Datasets	45
4.1.1	RAVDESS	46
4.1.2	CREMA-D	46
4.2	Data Processing	47
4.3	Model Evaluation	48
4.3.1	Classification Results	48
4.3.2	Descriptive Statistics	50
4.3.3	Out of Distribution Evaluation	51
4.3.4	Extension from Binary to Multi-Emotion Classification	52
4.4	Data Analysis	54
4.4.1	Frame Size Analysis	54
4.4.2	Time Complexity	55
4.4.3	Face-Region Occlusion Analysis	56
4.4.4	Facial Recognition and Actor Identification Experiment	59
4.4.5	Actor Identification with Rotation Analysis	60
5	Conclusion	63
	Bibliography	66

List of Tables

4.1	Summary of experimental analyses	45
4.2	Best observed test accuracy (%) across eight random seeds	49
4.3	Confusion matrix for the best RB-QCR run on RAVDESS (accuracy = 98.75%)	50
4.4	Confusion matrix for the best RB-QCR run on CREMA-D (accuracy = 96.63%)	50
4.5	Descriptive statistics of test accuracy across eight random seeds	50
4.6	RAVDESS–CREMA-D mixed-dataset split analysis for RB-QCR at fixed $\lambda =$ 10^{-4}	52
4.7	Classification accuracy under increasing emotion-set size	53
4.8	RB-QCR accuracy (%) under different frame-size configurations	54
4.9	Computational complexity of Algorithm 4 (RB-QCR).	56
4.10	Actor-identification accuracy using the reduced biquaternion representation .	59
4.11	Random per-clip rotation jitter accuracy for actor identification	61
4.12	Fixed-angle rotation accuracy for actor identification	62

List of Figures

4.1	Sample frames from the RAVDESS dataset showing actors displaying happy (left) and sad (right) expressions.	46
4.2	Sample frames from the CREMA-D dataset showing actors displaying happy (left) and sad (right) expressions.	47
4.3	Face-region occlusion sensitivity heatmap for RB-QCR. Each cell gives the accuracy drop after occluding the corresponding facial patch across all frames.	57
4.4	Occlusion sensitivity overlaid on sample test frames. The strongest drops occur around the mouth, cheeks, and lower facial regions.	58
4.5	Actor-ID accuracy under fixed rotations and random per-clip rotation jitter.	60
4.6	Example test-time rotations applied to a face crop. Larger rotations introduce substantial changes in alignment and visible facial geometry.	62

Chapter 1

Literature Review

Video classification is a central problem in modern machine learning with applications in security and surveillance, driver monitoring systems, sports analytics, and emotion recognition. This thesis focuses on emotion recognition, which aims to infer a person’s emotional state from a video of their face. The complexities of this problem manifest in the need to capture both the local spatial structure of facial expressions within individual frames and their temporal evolution across time, since emotional cues may vary in intensity and may not be consistently expressed throughout the video. As a result, some approaches perform frame-level analysis followed by temporal aggregation. For example, Lee et al. [6] estimated emotion scores at the frame level and used them to weight and aggregate frame features into a single video-level representation for final classification, rather than treating all frames equally. In contrast, this thesis adopts a simpler approach by uniformly sampling frames from each video and performing classification at the video level.

Deep learning is widely regarded as the state of the art in emotion recognition and video classification, and recent work has adapted it specifically to facial emotion on the datasets used in this thesis. Sotelo-Barrales et al. [15] combined facial-landmark geometry with compact features from convolutional networks and passed the resulting frame sequences to Long Short-Term Memory and bidirectional Long Short-Term Memory models, reporting competi-

tive accuracy on the Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D) and the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). Ryumina and Karpov [12] instead represented each face by the pairwise Euclidean distances between its landmarks, ranked those distances by ensemble feature-importance scores, and passed the most informative subset to a Long Short-Term Memory classifier; they tested their method both on a single dataset and across different datasets, although both methods are comparatively lightweight, they remain subject to the costs that characterize deep learning more broadly. As Selva et al. [14] document in their survey of video transformer models, such approaches typically estimate large numbers of parameters, rely on large labelled training sets, and require a substantial computational budget for training. These requirements limit their applicability in settings where labelled data are scarce or costly to obtain, where privacy or contractual constraints prohibit transferring data to external systems, or where training on dedicated infrastructure is infeasible.

These constraints motivate the use of lightweight classifiers. Regression-based models have historically served this role effectively in a range of settings, with ridge regression being a standard example. Representation-based classifiers in particular align well with this objective. Naseem et al. [8] introduced linear regression classification, in which a test image is expressed as a least-squares linear combination of the training images within each class and assigned to the class that yields the smallest reconstruction error, avoiding the need to learn a large parametric model.

Several extensions of this idea have demonstrated strong performance on challenging recognition tasks. Wright et al. [16] proposed sparse representation-based classification, improving robustness to occlusion and corruption in face recognition. Zhang et al. [20] later introduced collaborative representation, where all classes jointly contribute to the reconstruction, achieving comparable accuracy at significantly reduced computational cost. Because images and videos are naturally multi-dimensional, these regression principles have been extended to tensor settings, where predictors are represented as multi-way arrays rather than

vectors. Zhou et al. [21] developed a tensor regression framework for array-valued data, which extends naturally to video by treating time as an additional mode.

Linear regression-based classification in the real domain treats the red, green, and blue channels as independent values, hence discarding their intrinsic coupling. In contrast, Hamilton quaternions encode a colour pixel as a single hypercomplex entity, preserving inter-channel dependencies in a unified algebraic structure. Zou, Kou, and Wang [23] first extended sparse and collaborative representation classification to the quaternion domain, encoding each colour image as a quaternion vector and solving the resulting optimization problems through equivalent real-valued reformulations. Building on this framework, Zou et al. [24] developed quaternion-based linear regression classification for colour face recognition.

Reduced biquaternions, introduced by Pei et al. [9], provide a commutative alternative to Hamilton quaternions while maintaining a compact Fourier-transform structure, thereby reducing computational complexity. More recent work has extended these algebraic frameworks to higher-dimensional data analysis. Yuan et al. [18] proposed a direct solver for regularized quaternion matrix regression, while Yu et al. [17] developed a singular value decomposition for third-order reduced biquaternion tensors and demonstrated its effectiveness in applications such as fast colour video compression and deblurring.

This thesis combines two lines of work by extending linear regression-based classification from colour images to colour video, and by utilizing quaternion and reduced biquaternion representations for their established benefits. This provides an alternative approach for small-scale classification problems with a limited number of categories, where computational efficiency is important and lightweight models are preferred.

The main contributions of this thesis are as follows:

1. We extend quaternion linear regression (QLR) and quaternion collaborative representation (QCR) from static colour images to colour video by representing each clip as a third-order quaternion-valued tensor and deriving the associated regression-based classification procedures.
2. We develop the corresponding reduced biquaternion models, RB-QLR and RB-QCR, in which the idempotent decomposition reduces the regression problem to two complex-valued linear systems, avoiding the full Hamilton quaternion embedding and substantially lowering memory cost.
3. We benchmark all four models on publicly available datasets RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) CREMA-D (Crowd-sourced Emotional Multimodal Actors Dataset) under actor-independent train, validation, and test splits, together with a mixed cross-dataset evaluation. The results show that RB-QCR achieves the strongest classification accuracy while remaining tractable at frame resolutions where Hamilton quaternion QCR exceeds available memory.
4. We analyze RB-QCR through multiple experiments designed to evaluate its classification performance, examine its robustness under different settings, and assess how changes in model configuration affect its behaviour.

The remainder of this thesis develops and evaluates quaternion and reduced biquaternion regression-based classifiers for colour video emotion recognition. We extend quaternion linear regression (QLR) and quaternion collaborative representation (QCR) from static colour images to colour video, and derive the corresponding reduced biquaternion models (RB-QLR and RB-QCR), which decompose the regression problem into two independent complex-valued linear systems. All four classifiers are benchmarked on the RAVDESS and CREMA-D datasets, after which the best-performing model (RB-QCR) is evaluated in greater detail to assess its robustness and behaviour across multiple experimental settings.

Chapter 2 provides the necessary background for this thesis, introducing key concepts and briefly reviewing the historical development of classification, ridge regression, regression based classification, tensors, quaternions, and reduced biquaternions and their evolution toward modern image and video classification.

Chapter 3 presents the main theoretical contributions, including the development of the proposed models and the derivation of the associated proofs, and the algorithms to implement them.

Chapter 4 reports the experimental results, evaluating all four classifiers on benchmark emotion recognition datasets under both actor-independent and cross-dataset settings. It also provides a more detailed analysis of the RB-QCR model through a range of experiments examining its accuracy, robustness, and computational behaviour.

Finally, Chapter 5 concludes the thesis and outlines directions for future work.

Chapter 2

Preliminaries

This chapter introduces the mathematical and methodological foundations required for the development of the proposed quaternion and reduced biquaternion regression-based classifiers. Section 2.1 provides a brief overview of the classification task, and Section 2.2 reviews ridge regression, which forms the basis of the regression-based classification framework used throughout this thesis. Section 2.3 then discusses how regression can be adapted for classification then we explain the difference between collaborative representation and linear regression for classification. Section 2.4 introduces tensors, which provide a natural framework for representing video data, and Section 2.5 extends this to tensor regression. Section 2.6 presents the algebraic foundations of Hamilton quaternions, and Section 2.7 introduces reduced biquaternions, highlighting the properties that make both algebras well suited for colour video classification. Finally, Section 2.8 compares the two and outlines the computational advantages that reduced biquaternions offer over Hamilton quaternions.

2.1 Classification

Classification is a supervised learning task in which the goal is to learn a decision rule that assigns a discrete class label to a new, previously unseen input. Formally, let $\{(\mathbf{x}_i, \ell_i)\}_{i=1}^N$ denote a set of N labelled training examples, where $\mathbf{x}_i \in \mathbb{R}^d$ is the i -th input vector and

$\ell_i \in \{1, \dots, K\}$ is its corresponding class label, with K denoting the total number of classes. A classifier is then a function $f : \mathbb{R}^d \rightarrow \{1, \dots, K\}$ that maps an input vector to one of the K classes. Popular approaches include nearest-neighbour classifiers, discriminant analysis, and support vector machines, as well as, more recently, deep learning models.

2.2 Ridge Regression

Ridge regression is the regularized least-squares method at the core of the classifiers used in this thesis. Given a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ and a target vector $\mathbf{y} \in \mathbb{R}^m$, ordinary least squares seeks the coefficient vector $\boldsymbol{\beta}$ minimizing the reconstruction error $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$, where $\|\mathbf{v}\|_2 = (\sum_i v_i^2)^{1/2}$ denotes the ℓ_2 norm of a vector $\mathbf{v}$, so that $\|\cdot\|_2^2$ is the sum of squared entries. When the columns of $\mathbf{X}$ are correlated, the matrix $\mathbf{X}^\top \mathbf{X}$ becomes ill-conditioned or singular and the solution is unstable. To address this, Hoerl and Kennard [4] added an ℓ_2 penalty on the coefficients,

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2, \quad \lambda > 0,$$

which admits the closed-form solution

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X} + \lambda I)^{-1} \mathbf{X}^\top \mathbf{y}.$$

The penalty term guarantees that the inverse exists for any $\lambda > 0$, even when $\mathbf{X}^\top \mathbf{X}$ is singular. The regularization parameter λ controls the strength of the penalty, with larger values imposing stronger regularization to help reduce overfitting.

2.3 Linear Regression vs. Collaborative Representation

Although ridge regression predicts continuous outputs, it can be adapted for classification. Naseem et al. [8] introduced *linear regression classification* (LRC), which reconstructs the

test sample *separately within each class*: for every class, a least-squares fit is computed using only the training samples of that class, and the test sample is assigned to the class achieving the smallest reconstruction error. Building on this idea, Wright et al. [16] proposed sparse-representation-based classification, which instead codes the test sample jointly over the training samples of all classes by incorporating an ℓ_1 penalty into the optimization objective. The ℓ_1 penalty is defined as $\|\boldsymbol{\alpha}\|_1 = \sum_i |\alpha_i|$, where α_i denotes the i -th entry of the representation vector $\boldsymbol{\alpha}$, i.e., the coefficient assigned to the i -th training sample in the reconstruction.

Zhang et al. [20] then argued that it is the collaboration among classes, rather than the sparsity of the code, that drives recognition, and introduced *collaborative representation classification* (CRC). In contrast to LRC, CRC forms a *global* sample data matrix from the training samples of all classes, so that every class contributes simultaneously to the reconstruction of the test sample. The coefficients are obtained by solving a ridge-regularized least-squares problem of exactly the form introduced above, in which the ℓ_1 penalty is replaced by an ℓ_2 penalty, defined as $\|\boldsymbol{\alpha}\|_2^2 = \sum_i \alpha_i^2$, which penalizes the squared magnitudes of the coefficients rather than enforcing sparsity; the global coefficient vector is then restricted to the entries associated with each class, the corresponding class-wise residuals are computed, and the test sample is assigned to the class whose training samples reconstruct it with the smallest residual. The two models therefore share a single reconstruction framework and differ only in how the sample data matrix is formed, class-wise for LRC and global for CRC.

2.4 Tensors

A tensor is a multi-way array, generalizing scalars, vectors, and matrices to an arbitrary number of dimensions. A scalar is a tensor of order zero, a vector of order one, and a matrix of order two; arrays of order three or higher are called higher-order tensors. Formally, an order- N tensor is an element of $\mathbb{R}^{I_1 \times \dots \times I_N}$, where $I_1, \dots, I_N$ denote the sizes of the tensor along each of its N dimensions, and each dimension is called a *mode*. An entry is located

by giving one index per mode: the element of a third-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ in position (i_1, i_2, i_3) is written as $x_{i_1 i_2 i_3}$.

A greyscale image makes the idea concrete. It is a second-order tensor (a matrix) $\mathcal{X} \in \mathbb{R}^{H \times W}$, where H and W are the height and width in pixels and $x_{i_1 i_2}$ is the intensity of the pixel in row i_1 and column i_2 . Adding a third mode for the colour channels gives a colour image as a third-order tensor $\mathcal{X} \in \mathbb{R}^{H \times W \times 3}$, in which $x_{i_1 i_2 1}$, $x_{i_1 i_2 2}$, and $x_{i_1 i_2 3}$ are the red, green, and blue values of a single pixel. The value of this representation is that it keeps the natural layout of the data intact: the rows, columns, and channels remain separate, indexable axes, so the spatial arrangement of the pixels and the relationship between the colour channels are preserved rather than flattened into an unstructured vector.

Tensors originate in differential geometry and physics; their use in data analysis grew from the decomposition introduced by Hitchcock [3] and was later consolidated in the survey of Kolda and Bader [5]. This structure fits visual data directly, and video in particular. A colour video extends a colour image with a fourth, temporal mode, giving a fourth-order tensor $\mathcal{X} \in \mathbb{R}^{H \times W \times 3 \times T}$, where T is the number of frames. In this thesis, however, we represent each pixel as a single quaternion-valued entry that absorbs its red, green, and blue values, which removes the separate colour mode and lowers the order from four to three. Each colour clip is therefore a third-order quaternion tensor: the first two modes hold the height and width of each frame, and the third mode indexes the frames over time.

2.5 Tensor Regression

In tensor regression the predictor is a multi-way array rather than a vector, so for a clip represented as a third-order tensor the coefficients form a tensor of the same shape. Estimating them directly is costly, because the number of free parameters grows with the product of the mode dimensions. Two strategies address this. The first imposes low-rank structure on the coefficient tensor: Zhou et al. [21] developed such a framework for array-valued data, which keeps the parameter count in check while preserving the spatial layout of the input rather

than flattening it away. The second flattens the predictor and solves an ordinary regularized least-squares problem; the vectorization-based formulation of Zou et al. [24] for the matrix case extends directly to third-order tensors and is the route followed in this thesis.

2.6 Quaternions

Quaternions were introduced by Hamilton [2] as a four-dimensional extension of the complex numbers, with one real component and three imaginary components $\mathbf{i}, \mathbf{j}, \mathbf{k}$ [11]. The resulting algebra has developed into a mature area with applications in three-dimensional rotations, signal processing, and physics [11]. In image processing, Sangwine [13] first proposed encoding the red, green, and blue (RGB) channels of a colour pixel as the three imaginary components of a pure quaternion, which allows the three channels to be treated as a single algebraic entity rather than three independent scalar values. This representation has since been exploited in quaternion convolutional neural networks [22], quaternion linear regression classifiers for colour face recognition [24], and quaternion tensor completion for colour videos and images [7]. More recent work continues to leverage quaternion representations in autoencoders [1] and in direct solvers for regularized quaternion matrix regression [18].

We now introduce the algebraic preliminaries used throughout this thesis. The quaternion space $\mathbb{H}$ is defined as

$$\mathbb{H} = \{ q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \mid q_0, q_1, q_2, q_3 \in \mathbb{R} \},$$

where the imaginary units satisfy

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1, \quad \mathbf{ij} = \mathbf{k}, \quad \mathbf{jk} = \mathbf{i}, \quad \mathbf{ki} = \mathbf{j}.$$

Multiplication in $\mathbb{H}$ is noncommutative; that is, for $p, q \in \mathbb{H}$, $pq \neq qp$ in general. A quaternion is called a *pure quaternion* when its scalar part is zero, i.e., $q_0 = 0$. This is the form used to

represent RGB pixels throughout this thesis.

The conjugate of q is defined as

$$\bar{q} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k},$$

and the modulus (or norm) of q is

$$|q| = \sqrt{q\bar{q}} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}.$$

A quaternion vector $\mathbf{v} \in \mathbb{H}^n$ is a vector whose entries are quaternions,

$$\mathbf{v} = [v_1, \dots, v_n]^\top, \quad v_i \in \mathbb{H}.$$

The inner product between two quaternion vectors $\mathbf{u}, \mathbf{v} \in \mathbb{H}^n$ is defined as

$$\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^H \mathbf{v} = \sum_{i=1}^n \bar{u}_i v_i,$$

where $\mathbf{u}^H = \bar{\mathbf{u}}^\top$ denotes the conjugate transpose of $\mathbf{u}$. The induced ℓ_2 norm is then

$$\|\mathbf{v}\|_2 = \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle}.$$

2.7 Reduced Biquaternions

The reduced biquaternion algebra, denoted $\mathbb{H}_c$, is a four-dimensional hypercomplex algebra that resembles Hamilton quaternions. Unlike Hamilton quaternions, $\mathbb{H}_c$ is commutative and admits an orthogonal idempotent basis $\{\mathbf{e}_1, \mathbf{e}_2\}$ under which each reduced biquaternion decomposes into two independent complex components. Reduced biquaternions have been applied to colour image processing and signal analysis due to their algebraic properties making them an attractive alternative to Hamilton quaternions for colour-based classification

problems where computational efficiency is important [9, 10].

The reduced biquaternion algebra $\mathbb{H}_c$ is defined as

$$\mathbb{H}_c = \{ a_0 + a_1 \mathbf{i} + a_2 \mathbf{j} + a_3 \mathbf{k} \mid a_0, a_1, a_2, a_3 \in \mathbb{R} \},$$

where the imaginary units satisfy

$$\mathbf{i}^2 = -1, \quad \mathbf{j}^2 = 1, \quad \mathbf{ij} = \mathbf{ji} = \mathbf{k}.$$

It follows that

$$\mathbf{k}^2 = -1, \quad \mathbf{ki} = \mathbf{ik} = -\mathbf{j}, \quad \mathbf{jk} = \mathbf{kj} = \mathbf{i}.$$

Hence $\mathbb{H}_c$ is a four-dimensional commutative algebra, in contrast to Hamilton quaternions. Moreover, $\mathbb{H}_c$ contains zero divisors; for example, $(1 + \mathbf{j})(1 - \mathbf{j}) = 0$, so $1 + \mathbf{j}$ and $1 - \mathbf{j}$ are nonzero elements without inverses. We now introduce an alternative representation of reduced biquaternions that will be useful later. A reduced biquaternion $q \in \mathbb{H}_c$ can be written in the standard basis $\{1, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$ as

$$q = q_0 + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}, \quad q_0, q_1, q_2, q_3 \in \mathbb{R}.$$

Besides this standard form, we introduce an equivalent one using an idempotent orthogonal basis $\{\mathbf{e}_1, \mathbf{e}_2\}$. The use of such bases originates in ring theory and hypercomplex algebra, where idempotent elements are used to decompose an algebra into a direct sum of simpler subalgebras. A similar representation was introduced in [9] for the hypercomplex algebra HCA_4 , which is isomorphic to the reduced biquaternion algebra $\mathbb{H}_c$. We adopt the same notation here because it separates a reduced biquaternion into two complex components, which simplifies both algebraic manipulations and subsequent computations.

Define

$$\mathbf{e}_1 = \frac{1 + \mathbf{j}}{2}, \quad \mathbf{e}_2 = \frac{1 - \mathbf{j}}{2}. \tag{2.1}$$

Using $\mathbf{j}^2 = 1$, we obtain

$$\mathbf{e}_1^2 = \mathbf{e}_1, \quad \mathbf{e}_2^2 = \mathbf{e}_2, \quad \mathbf{e}_1\mathbf{e}_2 = 0, \quad \mathbf{e}_1 + \mathbf{e}_2 = 1. \quad (2.2)$$

Hence, $\mathbf{e}_1$ and $\mathbf{e}_2$ are orthogonal and idempotent.

We now derive the corresponding representation of q . Since

$$1 = \mathbf{e}_1 + \mathbf{e}_2, \quad \mathbf{j} = \mathbf{e}_1 - \mathbf{e}_2,$$

it follows that

$$\mathbf{k} = \mathbf{ij} = \mathbf{i}(\mathbf{e}_1 - \mathbf{e}_2) = \mathbf{ie}_1 - \mathbf{ie}_2.$$

Substituting these identities into

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k},$$

we get

$$\begin{aligned} q &= q_0(\mathbf{e}_1 + \mathbf{e}_2) + q_1\mathbf{i}(\mathbf{e}_1 + \mathbf{e}_2) + q_2(\mathbf{e}_1 - \mathbf{e}_2) + q_3\mathbf{i}(\mathbf{e}_1 - \mathbf{e}_2) \\ &= (q_0 + q_2)\mathbf{e}_1 + (q_0 - q_2)\mathbf{e}_2 + (q_1 + q_3)\mathbf{ie}_1 + (q_1 - q_3)\mathbf{ie}_2 \\ &= [(q_0 + q_2) + (q_1 + q_3)\mathbf{i}]\mathbf{e}_1 + [(q_0 - q_2) + (q_1 - q_3)\mathbf{i}]\mathbf{e}_2. \end{aligned}$$

Therefore, every reduced biquaternion can be written as

$$q = z_1\mathbf{e}_1 + z_2\mathbf{e}_2, \quad (2.3)$$

where

$$z_1 = (q_0 + q_2) + (q_1 + q_3)\mathbf{i}, \quad z_2 = (q_0 - q_2) + (q_1 - q_3)\mathbf{i}.$$

Since $q_0, q_1, q_2, q_3 \in \mathbb{R}$, both z_1 and z_2 are of the form $a + b\mathbf{i}$ with $a, b \in \mathbb{R}$. Hence $z_1, z_2 \in \mathbb{C}$.

The representation in (2.3) shows that the reduced biquaternion algebra decomposes into two complex-valued components, a property that will play a central role in the regression models developed later.

We now list some basic properties of reduced biquaternions.

(1) The conjugate of

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

is defined as

$$\bar{q} = q_0 - q_1\mathbf{i} + q_2\mathbf{j} - q_3\mathbf{k}.$$

Equivalently, if $q = z_1\mathbf{e}_1 + z_2\mathbf{e}_2$, then

$$\bar{q} = \overline{z_2}\mathbf{e}_1 + \overline{z_1}\mathbf{e}_2,$$

where the bar on z_1 and z_2 denotes complex conjugation.

(2) The norm of q is defined by

$$|q| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}.$$

In general,

$$|q|^2 \neq q\bar{q}.$$

(3) There are several ways to define the norm. In this thesis, we use the following definition:

$$\mathbf{u} = (u_1, \dots, u_m)^\top \in \mathbb{H}_c^m,$$

$$\|\mathbf{u}\|_2^2 = \sum_{r=1}^m |u_r|^2.$$

2.8 Reduced Biquaternions vs. Hamilton Quaternions

Having established the algebraic structure of $\mathbb{H}_c$, we now examine the computational advantages it offers over Hamilton quaternions. The key benefit arises from the idempotent decomposition (2.3), which reduces a reduced biquaternion product to two independent complex multiplications [9].

Let

$$q_1 = z_1 \mathbf{e}_1 + z_2 \mathbf{e}_2, \quad q_2 = w_1 \mathbf{e}_1 + w_2 \mathbf{e}_2.$$

Using the idempotent properties in (2.2), the mixed terms $\mathbf{e}_1 \mathbf{e}_2$ and $\mathbf{e}_2 \mathbf{e}_1$ vanish, and therefore

$$q_1 q_2 = (z_1 w_1) \mathbf{e}_1 + (z_2 w_2) \mathbf{e}_2. \tag{2.4}$$

Thus, the multiplication decouples into two independent complex multiplications.

To make the computational gain explicit, recall that a complex multiplication

$$(a + b\mathbf{i})(c + d\mathbf{i}) = (ac - bd) + (ad + bc)\mathbf{i}$$

requires four real multiplications: ac , bd , ad , and bc . Therefore, computing $z_1 w_1$ and $z_2 w_2$ requires a total of $2 \times 4 = 8$ real multiplications.

In contrast, performing multiplication directly in the standard four-component form

$$q = a_0 + a_1 \mathbf{i} + a_2 \mathbf{j} + a_3 \mathbf{k}$$

requires full interaction between all components, leading to $4 \times 4 = 16$ real multiplications and 12 real additions. Hence, the idempotent decomposition reduces the number of real multiplications per product from 16 to 8 [9]. This halving of arithmetic cost carries over to vector and matrix operations, making $\mathbb{H}_c$ an attractive setting for the regression models developed in the following sections.

Chapter 3

Main Results

In this chapter, we introduce the general regression-based classification framework that underlies all four models developed in this thesis. Both the Linear Regression Classifier (LRC) and the Collaborative Representation Classifier (CRC) share a common reconstruction problem: a test sample is approximated as a linear combination of training samples, and the sample is assigned to the class whose reconstruction yields the smallest residual. The two models differ only in the construction of the sample matrix and in whether regularization is applied. We start by developing the quaternion-based formulations (QCR and QLR) and then introduce their reduced biquaternion counterparts (RB-QCR and RB-QLR).

The chapter is organized as follows. In Section 3.1, we represent each RGB video as a third-order pure quaternion tensor, in which the three colour channels are encoded in the imaginary units, and we introduce a vectorization operator that maps each tensor to a quaternion vector. In Section 3.2, we collect the vectorized training samples into a single dictionary matrix and formulate the classification task as a reconstruction problem. Because quaternion multiplication is non-commutative, Section 3.3 introduces the linear operators $P(\cdot)$ and $Q(\cdot)$, which embed quaternion matrices and vectors into real-valued counterparts. Building on this reformulation, Section 3.4 derives Quaternion Collaborative Representation (QCR), which solves a single global ridge regression over all training samples, proves its

equivalence to a real-valued problem, and classifies by class-wise residuals, while Section 3.5 derives Quaternion Linear Regression (QLR), which fits class-specific models using either a regularized closed-form solution or the Moore–Penrose pseudoinverse. We then turn to the reduced biquaternion setting. Section 3.6 develops Reduced Biquaternion Collaborative Representation (RB-QCR): using the orthogonal idempotent decomposition, we prove that the reduced biquaternion ridge regression problem decouples into two independent complex ridge regressions with closed-form solutions, eliminating the need for the larger real-valued embeddings required in the quaternion case. Section 3.7 develops the class-wise RB-QLR model, which decomposes in the same way and admits both a regularized and a pseudoinverse-based solution. Finally, Section 3.8 presents the residual-based classification rule for the collaborative model.

3.1 Quaternion Tensor Representation and Vectorization

In this thesis, each video sample is represented as a third-order quaternion tensor, where the RGB colour channels are encoded in the $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ components.

For an RGB pixel with channel values (R, G, B) , we model it as a pure quaternion.

$$q = Ri + Gj + Bk, \quad R, G, B \in \mathbb{R},$$

Since only three colour channels (R, G, B) need to be encoded, and these map naturally onto the three imaginary units of the quaternion, no information remains for the real part, which is accordingly set to zero. Beyond being a natural encoding, this yields a *pure quaternion* representation that simplifies several of the algebraic operations used in the following sections. Note that the RGB information is encoded directly in the quaternion structure rather than stored as a separate channel dimension. Consequently, a video consisting of T frames

of spatial size $H \times W$ is represented as a third-order quaternion tensor

$$\mathcal{X} \in \mathbb{H}^{H \times W \times T},$$

where each entry of $\mathcal{X}$ encodes the RGB information of one pixel.

We now introduce a notation that will be used throughout the thesis. Let

$$\mathcal{X} \in \mathbb{H}^{H \times W \times T}, \quad \mathbf{X} \in \mathbb{H}^{m \times n}, \quad \mathbf{v} \in \mathbb{H}^N.$$

Any quaternion tensor

$$\mathcal{X} \in \mathbb{H}^{H \times W \times T}$$

admits the componentwise representation

$$\mathcal{X} = \mathcal{X}_0 + \mathcal{X}_1 \mathbf{i} + \mathcal{X}_2 \mathbf{j} + \mathcal{X}_3 \mathbf{k}, \quad \mathcal{X}_\ell \in \mathbb{R}^{H \times W \times T}, \quad \ell = 0, 1, 2, 3.$$

Similarly, a quaternion matrix $\mathbf{X} \in \mathbb{H}^{m \times n}$ can be written as

$$\mathbf{X} = \mathbf{X}_0 + \mathbf{X}_1 \mathbf{i} + \mathbf{X}_2 \mathbf{j} + \mathbf{X}_3 \mathbf{k}, \quad \mathbf{X}_\ell \in \mathbb{R}^{m \times n}, \quad \ell = 0, 1, 2, 3,$$

and a quaternion vector $\mathbf{v} \in \mathbb{H}^N$ can be written as

$$\mathbf{v} = \mathbf{v}_0 + \mathbf{v}_1 \mathbf{i} + \mathbf{v}_2 \mathbf{j} + \mathbf{v}_3 \mathbf{k}, \quad \mathbf{v}_\ell \in \mathbb{R}^N, \quad \ell = 0, 1, 2, 3.$$

Under the RGB encoding

$$q = R\mathbf{i} + G\mathbf{j} + B\mathbf{k}, \quad R, G, B \in \mathbb{R},$$

the quaternion coefficients satisfy

$$q_0 = 0, \quad q_1 = R, \quad q_2 = G, \quad q_3 = B.$$

For any third-order quaternion tensor $\mathcal{T} \in \mathbb{H}^{H \times W \times T}$ we define the vectorization operator

$$\text{Vec} : \mathbb{H}^{H \times W \times T} \rightarrow \mathbb{H}^{HWT}$$

by stacking the column-wise vectorizations of its frontal slices in order. Specifically, for each $t = 1, \dots, T$, the t -th frontal slice

$$\mathcal{T}(:, :, t) \in \mathbb{H}^{H \times W}$$

is a quaternion matrix representing the video image at frame t . We define

$$\text{Vec}(\mathcal{T}) = \begin{bmatrix} \text{vec}(\mathcal{T}(:, :, 1)) \\ \text{vec}(\mathcal{T}(:, :, 2)) \\ \vdots \\ \text{vec}(\mathcal{T}(:, :, T)) \end{bmatrix}.$$

Hence, for $h = 1, \dots, H$, $w = 1, \dots, W$, and $t = 1, \dots, T$, the entry

$$\mathcal{T}(h, w, t)$$

corresponds to the element

$$\text{Vec}(\mathcal{T})_k, \quad \text{where} \quad k = h + (w - 1)H + (t - 1)HW.$$

3.2 Problem Setup and Data Representation

Suppose we are given N quaternion-valued third-order tensors for classification, belonging to K distinct classes. Each sample is an element of $\mathbb{H}^{M \times W \times T}$, where M , W , and T denote the spatial height, spatial width, and number of frames, respectively.

Let $\mathcal{X} \in \mathbb{H}^{M \times W \times T}$ be a training sample, using the vectorization operator yields:

$$\mathbf{x}_i = \text{Vec}(\mathcal{X}_i) \in \mathbb{H}^{HWT}, \quad i = 1, \dots, N.$$

Collecting all vectorized training samples together, we define the global training matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_N \end{bmatrix} \in \mathbb{H}^{HWT \times N}.$$

If each column vector is written explicitly as

$$\mathbf{x}_j = \begin{bmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{HWT,j} \end{bmatrix}, \quad j = 1, \dots, N,$$

then $\mathbf{X}$ has the form

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{HWT,1} & x_{HWT,2} & \cdots & x_{HWT,N} \end{bmatrix}, \quad x_{ij} \in \mathbb{H}.$$

Similarly, given a test tensor $\mathbf{y} = \text{Vec}(\mathcal{Y}) \in \mathbb{H}^{HWT}$. The objective is to reconstruct the

test sample $\mathbf{y}$ as a linear combination of the training samples. Let

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix} \in \mathbb{H}^N$$

denote the coefficient vector. Then the reconstruction model is

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{v}.$$

Since

$$\mathbf{X} \in \mathbb{H}^{HWT \times N}, \quad \mathbf{v} \in \mathbb{H}^N,$$

it follows that

$$\mathbf{X}\mathbf{v} \in \mathbb{H}^{HWT},$$

that is, $\mathbf{X}\mathbf{v}$ is a quaternion-valued column vector with HWT entries. Equivalently, in entrywise form,

$$\mathbf{X}\mathbf{v} = \begin{bmatrix} x_{11}v_1 + x_{12}v_2 + \cdots + x_{1N}v_N \\ x_{21}v_1 + x_{22}v_2 + \cdots + x_{2N}v_N \\ \vdots \\ x_{HWT,1}v_1 + x_{HWT,2}v_2 + \cdots + x_{HWT,N}v_N \end{bmatrix}.$$

Hence, the test sample is approximated as

$$\mathbf{y} \approx \mathbf{X}\mathbf{v} = \mathbf{x}_1v_1 + \mathbf{x}_2v_2 + \cdots + \mathbf{x}_Nv_N.$$

The corresponding reconstruction error is therefore

$$\mathbf{e} = \mathbf{y} - \mathbf{X}\mathbf{v}.$$

Thus, the general problem is to determine a quaternion coefficient vector $\mathbf{v}$ such that the reconstruction $\mathbf{X}\mathbf{v}$ approximates the test sample $\mathbf{y}$ as accurately as possible.

3.3 Real-Valued Reformulation

In this section, we introduce two operators that enable us to exploit the advantages of quaternions while performing optimization in the real valued space. Since optimization in the quaternion space is complex because quaternion multiplication is non-commutative, the usual gradient-based and closed-form optimization tools do not transfer directly to the quaternion setting. It is therefore convenient to reformulate quaternion-valued problems through equivalent linear representations. Two common choices are to map each quaternion entry to a real-valued 4×4 matrix representation or a complex-valued 2×2 matrix representation.

Specifically, each quaternion

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

is mapped to the real-valued matrix

$$q \longmapsto \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \in \mathbb{R}^{4 \times 4}.$$

The equivalent complex-valued representation is

$$q \longmapsto \begin{bmatrix} q_0 + q_1i & q_2 + q_3i \\ -q_2 + q_3i & q_0 - q_1i \end{bmatrix} \in \mathbb{C}^{2 \times 2}.$$

To reformulate quaternion-valued vectors and matrices in real space, we further introduce

the linear operators $\mathbf{Q}$ and $\mathbf{P}$.

For a quaternion-valued vector

$$\mathbf{v} = v_0 + v_1\mathbf{i} + v_2\mathbf{j} + v_3\mathbf{k} \in \mathbb{H}^n,$$

where $v_0, v_1, v_2, v_3 \in \mathbb{R}^n$, define the column vector as follows:

$$\mathbf{Q}(\mathbf{v}) = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} \in \mathbb{R}^{4n}.$$

For a quaternion-valued matrix

$$\mathbf{X} = X_0 + X_1\mathbf{i} + X_2\mathbf{j} + X_3\mathbf{k} \in \mathbb{H}^{M \times N},$$

where $X_0, X_1, X_2, X_3 \in \mathbb{R}^{M \times N}$, define

$$\mathbf{P}(\mathbf{X}) = \begin{bmatrix} X_0 & -X_1 & -X_2 & -X_3 \\ X_1 & X_0 & -X_3 & X_2 \\ X_2 & X_3 & X_0 & -X_1 \\ X_3 & -X_2 & X_1 & X_0 \end{bmatrix} \in \mathbb{R}^{4M \times 4N}.$$

3.4 Quaternion Collaborative Representation (QCR)

Define the global dictionary by concatenating all training samples:

$$\mathbf{X} = [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(K)}] \in \mathbb{H}^{HWT \times N}, \quad N = \sum_{k=1}^K N_k.$$

Then QCR computes a single global coefficient vector

$$\mathbf{v}_{\text{QCR}} \in \mathbb{H}^N$$

by solving

$$\mathbf{v}_{\text{QCR}} = \arg \min_{\mathbf{v} \in \mathbb{H}^N} \|\mathbf{y} - \mathbf{X}\mathbf{v}\|_2^2 + \lambda \|\mathbf{v}\|_2^2, \quad \lambda > 0, \quad (\text{QCR})$$

which is equivalent to the following problem in $\mathbb{R}$:

$$\min_{\mathbf{v} \in \mathbb{H}^N} \|Q(\mathbf{y}) - P(\mathbf{X})Q(\mathbf{v})\|_2^2 + \lambda \|Q(\mathbf{v})\|_2^2. \quad (\text{QCR-R})$$

Now we prove the equivalency. For a quaternion vector

$$\mathbf{u} = u_0 + u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k} \in \mathbb{H}^m, \quad u_\ell \in \mathbb{R}^m,$$

we define

$$\|\mathbf{u}\|_2^2 := \sum_{r=1}^m |u_r|^2 = \|u_0\|_2^2 + \|u_1\|_2^2 + \|u_2\|_2^2 + \|u_3\|_2^2.$$

Moreover, since Q simply stacks the four real component vectors of $\mathbf{u}$ into a single real vector, it does not introduce any new values or alter the existing ones; it merely rearranges the same $4m$ real components already contained in $\mathbf{u}$. Hence,

$$Q(\mathbf{u}) = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{bmatrix},$$

and therefore

$$\|Q(\mathbf{u})\|_2^2 = \|u_0\|_2^2 + \|u_1\|_2^2 + \|u_2\|_2^2 + \|u_3\|_2^2 = \|\mathbf{u}\|_2^2. \quad (3.1)$$

Now we use the quaternion-valued matrix decomposition introduced in the previous sec-

tion.

$$\mathbf{X} = X_0 + X_1\mathbf{i} + X_2\mathbf{j} + X_3\mathbf{k}, \quad X_\ell \in \mathbb{R}^{m \times N},$$

and

$$\mathbf{v} = v_0 + v_1\mathbf{i} + v_2\mathbf{j} + v_3\mathbf{k}, \quad v_\ell \in \mathbb{R}^N.$$

Then

$$\begin{aligned} \mathbf{X}\mathbf{v} &= (X_0 + X_1\mathbf{i} + X_2\mathbf{j} + X_3\mathbf{k})(v_0 + v_1\mathbf{i} + v_2\mathbf{j} + v_3\mathbf{k}) \\ &= (X_0v_0 - X_1v_1 - X_2v_2 - X_3v_3) \\ &\quad + (X_1v_0 + X_0v_1 - X_3v_2 + X_2v_3)\mathbf{i} \\ &\quad + (X_2v_0 + X_3v_1 + X_0v_2 - X_1v_3)\mathbf{j} \\ &\quad + (X_3v_0 - X_2v_1 + X_1v_2 + X_0v_3)\mathbf{k}. \end{aligned}$$

Applying Q to both sides gives

$$Q(\mathbf{X}\mathbf{v}) = \begin{bmatrix} X_0v_0 - X_1v_1 - X_2v_2 - X_3v_3 \\ X_1v_0 + X_0v_1 - X_3v_2 + X_2v_3 \\ X_2v_0 + X_3v_1 + X_0v_2 - X_1v_3 \\ X_3v_0 - X_2v_1 + X_1v_2 + X_0v_3 \end{bmatrix}.$$

On the other hand, by the definitions of P and Q ,

$$P(\mathbf{X}) = \begin{bmatrix} X_0 & -X_1 & -X_2 & -X_3 \\ X_1 & X_0 & -X_3 & X_2 \\ X_2 & X_3 & X_0 & -X_1 \\ X_3 & -X_2 & X_1 & X_0 \end{bmatrix}, \quad Q(\mathbf{v}) = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix}.$$

Hence

$$P(\mathbf{X})Q(\mathbf{v}) = \begin{bmatrix} X_0v_0 - X_1v_1 - X_2v_2 - X_3v_3 \\ X_1v_0 + X_0v_1 - X_3v_2 + X_2v_3 \\ X_2v_0 + X_3v_1 + X_0v_2 - X_1v_3 \\ X_3v_0 - X_2v_1 + X_1v_2 + X_0v_3 \end{bmatrix}.$$

Therefore,

$$Q(\mathbf{X}\mathbf{v}) = P(\mathbf{X})Q(\mathbf{v}). \quad (3.2)$$

Now we prove that $Q(\mathbf{y} - \mathbf{X}\mathbf{v}) = Q(\mathbf{y}) - Q(\mathbf{X}\mathbf{v})$. Let

$$\mathbf{y} = y_0 + y_1\mathbf{i} + y_2\mathbf{j} + y_3\mathbf{k}, \quad \mathbf{X}\mathbf{v} = z_0 + z_1\mathbf{i} + z_2\mathbf{j} + z_3\mathbf{k},$$

where $y_\ell, z_\ell \in \mathbb{R}^m$ for $\ell = 0, 1, 2, 3$. Then

$$\mathbf{y} - \mathbf{X}\mathbf{v} = (y_0 - z_0) + (y_1 - z_1)\mathbf{i} + (y_2 - z_2)\mathbf{j} + (y_3 - z_3)\mathbf{k}.$$

Applying Q , we obtain

$$Q(\mathbf{y} - \mathbf{X}\mathbf{v}) = \begin{bmatrix} y_0 - z_0 \\ y_1 - z_1 \\ y_2 - z_2 \\ y_3 - z_3 \end{bmatrix}.$$

Since subtraction is taken componentwise in $\mathbb{R}^m$,

$$\begin{bmatrix} y_0 - z_0 \\ y_1 - z_1 \\ y_2 - z_2 \\ y_3 - z_3 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} - \begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \end{bmatrix} = Q(\mathbf{y}) - Q(\mathbf{X}\mathbf{v}).$$

Using the previous result and (3.2), it follows that

$$Q(\mathbf{y} - \mathbf{X}\mathbf{v}) = Q(\mathbf{y}) - P(\mathbf{X})Q(\mathbf{v}). \quad (3.3)$$

By applying (3.1), (3.2) and (3.3), we obtain

$$\begin{aligned} \|\mathbf{y} - \mathbf{X}\mathbf{v}\|_2^2 + \lambda\|\mathbf{v}\|_2^2 &= \|Q(\mathbf{y} - \mathbf{X}\mathbf{v})\|_2^2 + \lambda\|Q(\mathbf{v})\|_2^2 \\ &= \|Q(\mathbf{y}) - P(\mathbf{X})Q(\mathbf{v})\|_2^2 + \lambda\|Q(\mathbf{v})\|_2^2. \end{aligned} \quad (*)$$

Therefore, the quaternion problem is equivalent to minimizing its real-valued counterpart. The least-squares estimator is obtained by minimizing the residual with an ℓ_2 penalty. Setting the gradient of $\|Q(\mathbf{y}) - P(\mathbf{X})Q(\mathbf{v})\|_2^2 + \lambda\|Q(\mathbf{v})\|_2^2$ with respect to $Q(\mathbf{v})$ to zero yields the normal equation

$$(P(\mathbf{X})^\top P(\mathbf{X}) + \lambda I)Q(\mathbf{v}) = P(\mathbf{X})^\top Q(\mathbf{y}).$$

Since $\lambda > 0$, the matrix

$$P(\mathbf{X})^\top P(\mathbf{X}) + \lambda I$$

is symmetric positive definite and therefore invertible. Hence the unique minimizer is

$$Q(\mathbf{v}_{\text{QCR}}) = (P(\mathbf{X})^\top P(\mathbf{X}) + \lambda I)^{-1} P(\mathbf{X})^\top Q(\mathbf{y}). \quad (3.4)$$

Applying Q^{-1} , we recover the quaternion coefficient vector:

$$\mathbf{v}_{\text{QCR}} = Q^{-1} \left((P(\mathbf{X})^\top P(\mathbf{X}) + \lambda I)^{-1} P(\mathbf{X})^\top Q(\mathbf{y}) \right). \quad (3.5)$$

Now, we turn to the classification problem. After computing the global QCR coefficient vector $\mathbf{v}_{\text{QCR}}$, we evaluate the contribution of each class separately and assign the test sample to the class with the smallest reconstruction residual. Let $\mathcal{I}_k$ denote the index set of training

samples belonging to class k . Define the selection operator

$$\delta_k : \mathbb{H}^N \rightarrow \mathbb{H}^N, \quad \delta_k(\mathbf{v})_i = \begin{cases} v_i, & i \in \mathcal{I}_k, \\ 0, & i \notin \mathcal{I}_k. \end{cases}$$

Applying δ_k to the global coefficient vector yields the class-specific coefficients

$$\mathbf{v}_{\text{QCR}}^{(k)} = \delta_k(\mathbf{v}_{\text{QCR}}).$$

The class- k reconstruction of the test sample is then

$$\hat{\mathbf{y}}^{(k)} = \mathbf{X} \mathbf{v}_{\text{QCR}}^{(k)} = \mathbf{X} \delta_k(\mathbf{v}_{\text{QCR}})$$

and the corresponding residual is

$$r_k(\mathbf{y}) = \left\| \mathbf{y} - \mathbf{X} \mathbf{v}_{\text{QCR}}^{(k)} \right\|_2^2. \quad (3.6)$$

The final classification rule assigns $\mathbf{y}$ to the class with the smallest residual:

$$k^* = \arg \min_{k \in \{1, \dots, K\}} r_k(\mathbf{y}). \quad (3.7)$$

Algorithm 1 Quaternion Collaborative Representation Classification (QCR)

Input: Training matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{H}^{d \times N}$; a quaternion input vector $\mathbf{y} \in \mathbb{H}^d$; regularization parameter $\lambda > 0$.

Output: Predicted class label $\hat{k}$.

1. Normalize the columns of $\mathbf{X}$ to unit ℓ_2 -norm.
2. Compute $P(\mathbf{X})$ and $Q(\mathbf{y})$.
3. Solve for the global QCR coefficients

$$Q(\mathbf{v}_{\text{QCR}}) = (P(\mathbf{X})^\top P(\mathbf{X}) + \lambda I)^{-1} P(\mathbf{X})^\top Q(\mathbf{y}),$$

and recover $\mathbf{v}_{\text{QCR}} = Q^{-1}(Q(\mathbf{v}_{\text{QCR}}))$.

4. For each class $k = 1, \dots, K$, extract $\mathbf{v}_{\text{QCR}}^{(k)} = \delta_k(\mathbf{v}_{\text{QCR}})$.
 5. Compute the class residuals $r_k(\mathbf{y}) = \|\mathbf{y} - \mathbf{X}\mathbf{v}_{\text{QCR}}^{(k)}\|_2^2$.
 6. Classify: $\hat{k} = \arg \min_{k \in \{1, \dots, K\}} r_k(\mathbf{y})$.
-

3.5 Quaternion Linear Regression (QLR)

In contrast to QCR, Quaternion Linear Regression (QLR) reconstructs the test sample using training samples from each class separately. For each class $k \in \{1, \dots, K\}$, define the class-specific dictionary matrix

$$\mathbf{X}^{(k)} \in \mathbb{H}^{HWT \times N_k}.$$

For a given test vector $\mathbf{y} \in \mathbb{H}^{HWT}$, QLR computes a coefficient vector $\mathbf{v}_{\text{QLR}}^{(k)} \in \mathbb{H}^{N_k}$ by solving the least squares problem

$$\mathbf{v}_{\text{QLR}}^{(k)} = \arg \min_{\mathbf{v} \in \mathbb{H}^{N_k}} \|\mathbf{y} - \mathbf{X}^{(k)}\mathbf{v}\|_2^2. \quad (\text{QLR})$$

Using the linear operators Q and P introduced previously, problem (QLR) is equivalent

to the real-valued problem

$$\min_{\mathbf{v} \in \mathbb{H}^{N_k}} \|Q(\mathbf{y}) - P(\mathbf{X}^{(k)}) Q(\mathbf{v})\|_2^2. \quad (\text{QLR-R})$$

Indeed, the argument is identical to that of the QCR case with the penalty term omitted. Combining the norm preservation of Q in (3.1) with identities (3.2) and (3.3) applied to $\mathbf{X}^{(k)}$ gives

$$\|\mathbf{y} - \mathbf{X}^{(k)} \mathbf{v}\|_2^2 = \|Q(\mathbf{y} - \mathbf{X}^{(k)} \mathbf{v})\|_2^2 = \|Q(\mathbf{y}) - P(\mathbf{X}^{(k)}) Q(\mathbf{v})\|_2^2 \quad \text{for all } \mathbf{v} \in \mathbb{H}^{N_k}.$$

Since the regularization parameter λ need not be specified in advance, we consider two cases.

(1) Regularized solution Introducing a regularization parameter $\lambda > 0$, as in ridge regression, the derivation of the QCR case applies verbatim with $\mathbf{X}$ replaced by $\mathbf{X}^{(k)}$, yielding the unique minimizer

$$Q(\mathbf{v}_{\text{QLR}}^{(k)}) = (P(\mathbf{X}^{(k)})^\top P(\mathbf{X}^{(k)}) + \lambda I)^{-1} P(\mathbf{X}^{(k)})^\top Q(\mathbf{y}). \quad (3.8)$$

(2) Moore–Penrose pseudoinverse Alternatively, for $\lambda = 0$,

$$Q(\mathbf{v}_{\text{QLR}}^{(k)}) = P(\mathbf{X}^{(k)})^+ Q(\mathbf{y}), \quad (3.9)$$

which gives the minimal-norm least-squares solution via the real representation and the Moore–Penrose inverse [19]. When $P(\mathbf{X}^{(k)})$ has full column rank,

$$P(\mathbf{X}^{(k)})^+ = (P(\mathbf{X}^{(k)})^\top P(\mathbf{X}^{(k)}))^{-1} P(\mathbf{X}^{(k)})^\top,$$

so

$$Q(\mathbf{v}_{\text{QLR}}^{(k)}) = (P(\mathbf{X}^{(k)})^T P(\mathbf{X}^{(k)}))^{-1} P(\mathbf{X}^{(k)})^T Q(\mathbf{y}).$$

Hence,

$$\mathbf{v}_{\text{QLR}}^{(k)} = Q^{-1}(P(\mathbf{X}^{(k)})^+ Q(\mathbf{y})).$$

The class-specific reconstruction is

$$\hat{\mathbf{y}}^{(k)} = \mathbf{X}^{(k)} \mathbf{v}_{\text{QLR}}^{(k)},$$

and the corresponding residual is

$$r_k(\mathbf{y}) = \left\| \mathbf{y} - \mathbf{X}^{(k)} \mathbf{v}_{\text{QLR}}^{(k)} \right\|_2^2. \quad (3.10)$$

We note that when $\lambda = 0$ the minimizer of (QLR) need not be unique, but all minimizers attain the same residual value $r_k(\mathbf{y})$; the classification rule below is therefore well defined regardless of which solution is selected.

The final classification rule assigns $\mathbf{y}$ to the class with the smallest reconstruction error:

$$k^* = \arg \min_{k \in \{1, \dots, K\}} r_k(\mathbf{y}). \quad (3.11)$$

Algorithm 2 Quaternion Linear Regression Classification (QLR)

Input: Class-wise training matrices $\mathbf{X}^{(k)} \in \mathbb{H}^{d \times N_k}$, $k = 1, \dots, K$; a quaternion input vector $\mathbf{y} \in \mathbb{H}^d$; regularization parameter $\lambda \geq 0$.

Output: Predicted class label $\hat{k}$.

1. Normalize the columns of each $\mathbf{X}^{(k)}$.
2. Compute $P(\mathbf{X}^{(k)})$ for each class, and $Q(\mathbf{y})$.
3. For each class $k = 1, \dots, K$, compute the coefficients using either

$$Q(\mathbf{v}_{\text{QLR}}^{(k)}) = (P(\mathbf{X}^{(k)})^\top P(\mathbf{X}^{(k)}) + \lambda I)^{-1} P(\mathbf{X}^{(k)})^\top Q(\mathbf{y}) \quad (\text{if } \lambda > 0)$$

or

$$Q(\mathbf{v}_{\text{QLR}}^{(k)}) = P(\mathbf{X}^{(k)})^+ Q(\mathbf{y}) \quad (\text{if } \lambda = 0),$$

and recover $\mathbf{v}_{\text{QLR}}^{(k)} = Q^{-1}(Q(\mathbf{v}_{\text{QLR}}^{(k)}))$.

4. Compute the residuals $r_k(\mathbf{y}) = \|\mathbf{y} - \mathbf{X}^{(k)} \mathbf{v}_{\text{QLR}}^{(k)}\|_2^2$.
 5. Classify: $\hat{k} = \arg \min_{k \in \{1, \dots, K\}} r_k(\mathbf{y})$.
-

3.6 Reduced Biquaternion Collaborative Representation (RB-QCR)

Having developed the quaternion-based classifiers, we now turn to their reduced biquaternion counterparts. The key advantage of the reduced biquaternion algebra $\mathbb{H}_c$ is that the orthogonal idempotent decomposition reduces ridge regression to a pair of independent complex ridge regressions with closed-form solutions, eliminating the need for the larger $P(\cdot)$ and $Q(\cdot)$ real-valued embeddings used in the quaternion case.

We start by exploiting property (2.3) by substituting the channel-wise coefficients into the expressions for z_1 and z_2 , and noting that, since each pixel is modelled as a pure quaternion, we have $q_0 = 0$, with the red, green, and blue channels encoded in q_1 , q_2 , and q_3 , respectively.

$$z_1 = G + (R + B)i, \quad z_2 = -G + (R - B)i.$$

Applying this decomposition entrywise yields the corresponding complex-pair form for

tensors, matrices, and vectors. In particular,

$$\begin{aligned}\mathcal{X} &= \mathcal{X}_1 \mathbf{e}_1 + \mathcal{X}_2 \mathbf{e}_2, & \mathcal{X}_1, \mathcal{X}_2 &\in \mathbb{C}^{H \times W \times T}, \\ \mathbf{X} &= \mathbf{X}_1 \mathbf{e}_1 + \mathbf{X}_2 \mathbf{e}_2, & \mathbf{X}_1, \mathbf{X}_2 &\in \mathbb{C}^{m \times n}, \\ \mathbf{v} &= \mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2, & \mathbf{v}_1, \mathbf{v}_2 &\in \mathbb{C}^N.\end{aligned}\tag{3.12}$$

The collaborative representation model assumes that the test sample $\mathbf{y}$ can be approximately represented as a linear combination of all training samples

$$\mathbf{y} \approx \mathbf{X}\mathbf{v} = v_1 \mathbf{x}_1 + \cdots + v_N \mathbf{x}_N, \quad \mathbf{v} \in \mathbb{H}_c^N$$

and from (3.12), the product $\mathbf{X}\mathbf{v}$ separates as

$$\mathbf{X}\mathbf{v} = (\mathbf{X}_1 \mathbf{v}_1) \mathbf{e}_1 + (\mathbf{X}_2 \mathbf{v}_2) \mathbf{e}_2,$$

where $\mathbf{v}$ is the RB coefficient vector to be estimated. The estimation error ε is therefore measured by

$$\varepsilon = \|\mathbf{y} - \mathbf{X}\mathbf{v}\|.$$

To obtain an estimate of the coefficient vector, we minimize the reconstruction error using the classical ridge regression formulation. Given a matrix $\mathbf{X}$ and a test sample $\mathbf{y}$, ridge regression estimates the coefficient vector by minimizing the squared reconstruction error together with an ℓ_2 -regularization term:

$$\mathbf{v}^* = \arg \min_{\mathbf{v} \in \mathbb{H}_c^N} \|\mathbf{y} - \mathbf{X}\mathbf{v}\|_2^2 + \lambda \|\mathbf{v}\|_2^2. \tag{RB-QCR}$$

We now prove that the reduced biquaternion optimization problem (RB-QCR) is equivalent to two complex ridge regression problems.

Using the idempotent representation in (3.12), write

$$\mathbf{y} = \mathbf{y}_1 \mathbf{e}_1 + \mathbf{y}_2 \mathbf{e}_2, \quad \mathbf{X} = \mathbf{X}_1 \mathbf{e}_1 + \mathbf{X}_2 \mathbf{e}_2, \quad \mathbf{v} = \mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2,$$

where

$$\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{C}^{HWT}, \quad \mathbf{X}_1, \mathbf{X}_2 \in \mathbb{C}^{HWT \times N}, \quad \mathbf{v}_1, \mathbf{v}_2 \in \mathbb{C}^N.$$

Since

$$\mathbf{e}_1^2 = \mathbf{e}_1, \quad \mathbf{e}_2^2 = \mathbf{e}_2, \quad \mathbf{e}_1 \mathbf{e}_2 = \mathbf{e}_2 \mathbf{e}_1 = 0,$$

we have

$$\begin{aligned} \mathbf{X}\mathbf{v} &= (\mathbf{X}_1 \mathbf{e}_1 + \mathbf{X}_2 \mathbf{e}_2)(\mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2) \\ &= \mathbf{X}_1 \mathbf{v}_1 \mathbf{e}_1^2 + \mathbf{X}_1 \mathbf{v}_2 \mathbf{e}_1 \mathbf{e}_2 + \mathbf{X}_2 \mathbf{v}_1 \mathbf{e}_2 \mathbf{e}_1 + \mathbf{X}_2 \mathbf{v}_2 \mathbf{e}_2^2 \\ &= (\mathbf{X}_1 \mathbf{v}_1) \mathbf{e}_1 + (\mathbf{X}_2 \mathbf{v}_2) \mathbf{e}_2. \end{aligned}$$

Therefore

$$\mathbf{y} - \mathbf{X}\mathbf{v} = (\mathbf{y}_1 - \mathbf{X}_1 \mathbf{v}_1) \mathbf{e}_1 + (\mathbf{y}_2 - \mathbf{X}_2 \mathbf{v}_2) \mathbf{e}_2.$$

Next, we expand the norm carefully. Let

$$\mathbf{u} = \mathbf{u}_1 \mathbf{e}_1 + \mathbf{u}_2 \mathbf{e}_2 \in \mathbb{H}_c^m, \quad \mathbf{u}_1, \mathbf{u}_2 \in \mathbb{C}^m.$$

Write the r -th entry of $\mathbf{u}$ as

$$u_r = a_r + b_r \mathbf{i} + c_r \mathbf{j} + d_r \mathbf{k}, \quad a_r, b_r, c_r, d_r \in \mathbb{R}.$$

From the idempotent decomposition in section 2,

$$u_r = z_{r1} \mathbf{e}_1 + z_{r2} \mathbf{e}_2,$$

where

$$z_{r1} = (a_r + c_r) + (b_r + d_r)\mathbf{i}, \quad z_{r2} = (a_r - c_r) + (b_r - d_r)\mathbf{i}.$$

Since $z_{r1}, z_{r2} \in \mathbb{C}$, we use the standard complex norm,

$$|x + y\mathbf{i}|^2 = x^2 + y^2, \quad x, y \in \mathbb{R}.$$

Hence

$$|z_{r1}|^2 = (a_r + c_r)^2 + (b_r + d_r)^2,$$

$$|z_{r2}|^2 = (a_r - c_r)^2 + (b_r - d_r)^2.$$

Adding these two equalities gives

$$\begin{aligned} |z_{r1}|^2 + |z_{r2}|^2 &= (a_r + c_r)^2 + (a_r - c_r)^2 + (b_r + d_r)^2 + (b_r - d_r)^2 \\ &= 2a_r^2 + 2c_r^2 + 2b_r^2 + 2d_r^2 \\ &= 2(a_r^2 + b_r^2 + c_r^2 + d_r^2). \end{aligned}$$

Since the reduced biquaternion norm in Section 2 is

$$|u_r|^2 = a_r^2 + b_r^2 + c_r^2 + d_r^2,$$

Therefore,

$$|u_r|^2 = \frac{1}{2}(|z_{r1}|^2 + |z_{r2}|^2).$$

Summing both sides over $r = 1, \dots, m$, we get

$$\sum_{r=1}^m |u_r|^2 = \frac{1}{2} \sum_{r=1}^m (|z_{r1}|^2 + |z_{r2}|^2).$$

By linearity of summation,

$$\sum_{r=1}^m |u_r|^2 = \frac{1}{2} \left(\sum_{r=1}^m |z_{r1}|^2 + \sum_{r=1}^m |z_{r2}|^2 \right).$$

Since

$$\|\mathbf{u}\|_2^2 = \sum_{r=1}^m |u_r|^2, \quad \|\mathbf{u}_1\|_2^2 = \sum_{r=1}^m |z_{r1}|^2, \quad \|\mathbf{u}_2\|_2^2 = \sum_{r=1}^m |z_{r2}|^2,$$

it follows that

$$\|\mathbf{u}\|_2^2 = \frac{1}{2} (\|\mathbf{u}_1\|_2^2 + \|\mathbf{u}_2\|_2^2). \quad (3.13)$$

We now apply (3.13) first to the residual vector

$$\mathbf{y} - \mathbf{X}\mathbf{v} = (\mathbf{y}_1 - \mathbf{X}_1\mathbf{v}_1)\mathbf{e}_1 + (\mathbf{y}_2 - \mathbf{X}_2\mathbf{v}_2)\mathbf{e}_2.$$

Then the objective function becomes the following:

$$\begin{aligned} \|\mathbf{y} - \mathbf{X}\mathbf{v}\|_2^2 + \lambda\|\mathbf{v}\|_2^2 &= \frac{1}{2} (\|\mathbf{y}_1 - \mathbf{X}_1\mathbf{v}_1\|_2^2 + \|\mathbf{y}_2 - \mathbf{X}_2\mathbf{v}_2\|_2^2) \\ &\quad + \frac{\lambda}{2} (\|\mathbf{v}_1\|_2^2 + \|\mathbf{v}_2\|_2^2) \\ &= \frac{1}{2} \left(\|\mathbf{y}_1 - \mathbf{X}_1\mathbf{v}_1\|_2^2 + \lambda\|\mathbf{v}_1\|_2^2 \right) \\ &\quad + \frac{1}{2} \left(\|\mathbf{y}_2 - \mathbf{X}_2\mathbf{v}_2\|_2^2 + \lambda\|\mathbf{v}_2\|_2^2 \right). \end{aligned}$$

Now define

$$A(\mathbf{v}_1) = \|\mathbf{y}_1 - \mathbf{X}_1\mathbf{v}_1\|_2^2 + \lambda\|\mathbf{v}_1\|_2^2,$$

and

$$B(\mathbf{v}_2) = \|\mathbf{y}_2 - \mathbf{X}_2\mathbf{v}_2\|_2^2 + \lambda\|\mathbf{v}_2\|_2^2.$$

Then the full objective is

$$\frac{1}{2}A(\mathbf{v}_1) + \frac{1}{2}B(\mathbf{v}_2).$$

Since $\mathbf{v}_1$ appears only in A and $\mathbf{v}_2$ appears only in B , minimizing the full objective over $(\mathbf{v}_1, \mathbf{v}_2)$ is equivalent to minimizing A with respect to $\mathbf{v}_1$ and minimizing B with respect to $\mathbf{v}_2$ separately. Therefore,

$$\min_{\mathbf{v}_\ell \in \mathbb{C}^N} \|\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell\|_2^2 + \lambda \|\mathbf{v}_\ell\|_2^2, \quad \ell = 1, 2. \quad (3.14)$$

This proves that the reduced biquaternion representation problem is equivalent to two complex ridge regression problems.

We now derive the closed-form solution of (3.14). Fix $\ell \in \{1, 2\}$. We must minimize

$$\|\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell\|_2^2 + \lambda \|\mathbf{v}_\ell\|_2^2.$$

First expand the first term:

$$\begin{aligned} \|\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell\|_2^2 &= (\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell)^H (\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell) \\ &= \mathbf{y}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{y}_\ell + \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell. \end{aligned}$$

Adding the regularization term gives

$$\begin{aligned} \|\mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell\|_2^2 + \lambda \|\mathbf{v}_\ell\|_2^2 &= \mathbf{y}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{y}_\ell \\ &\quad + \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell + \lambda \mathbf{v}_\ell^H \mathbf{v}_\ell \\ &= \mathbf{y}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{y}_\ell \\ &\quad + \mathbf{v}_\ell^H (\mathbf{X}_\ell^H \mathbf{X}_\ell + \lambda \mathbf{I}) \mathbf{v}_\ell. \end{aligned}$$

Set

$$\mathbf{A}_\ell = \mathbf{X}_\ell^H \mathbf{X}_\ell + \lambda \mathbf{I}.$$

We now show that $\mathbf{A}_\ell$ is invertible. Let $\mathbf{z} \in \mathbb{C}^N$, $\mathbf{z} \neq \mathbf{0}$. Then

$$\begin{aligned}\mathbf{z}^H \mathbf{A}_\ell \mathbf{z} &= \mathbf{z}^H \mathbf{X}_\ell^H \mathbf{X}_\ell \mathbf{z} + \lambda \mathbf{z}^H \mathbf{z} \\ &= (\mathbf{X}_\ell \mathbf{z})^H (\mathbf{X}_\ell \mathbf{z}) + \lambda \|\mathbf{z}\|_2^2 \\ &= \|\mathbf{X}_\ell \mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_2^2.\end{aligned}$$

Since $\lambda > 0$ and $\mathbf{z} \neq \mathbf{0}$, we have $\|\mathbf{z}\|_2^2 > 0$, so

$$\mathbf{z}^H \mathbf{A}_\ell \mathbf{z} > 0.$$

Hence $\mathbf{A}_\ell$ is Hermitian positive definite, and therefore invertible.

Now we complete the square explicitly. Consider

$$(\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell)^H \mathbf{A}_\ell (\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell).$$

Expanding this product gives

$$\mathbf{v}_\ell^H \mathbf{A}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{A}_\ell \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell - (\mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell)^H \mathbf{A}_\ell \mathbf{v}_\ell + (\mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell)^H \mathbf{A}_\ell (\mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell).$$

Because $\mathbf{A}_\ell \mathbf{A}_\ell^{-1} = \mathbf{I}$, this becomes

$$\mathbf{v}_\ell^H \mathbf{A}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell + \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell.$$

Therefore

$$\begin{aligned}&\mathbf{v}_\ell^H \mathbf{A}_\ell \mathbf{v}_\ell - \mathbf{v}_\ell^H \mathbf{X}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{v}_\ell \\ &= (\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell)^H \mathbf{A}_\ell (\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell) - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell.\end{aligned}$$

Substituting this into the expanded objective, we obtain

$$\begin{aligned}
 & \| \mathbf{y}_\ell - \mathbf{X}_\ell \mathbf{v}_\ell \|_2^2 + \lambda \| \mathbf{v}_\ell \|_2^2 \\
 &= (\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell)^H \mathbf{A}_\ell (\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell) \\
 &+ \mathbf{y}_\ell^H \mathbf{y}_\ell - \mathbf{y}_\ell^H \mathbf{X}_\ell \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell.
 \end{aligned}$$

The second and third terms are constants with respect to $\mathbf{v}_\ell$. The first term is always nonnegative because $\mathbf{A}_\ell$ is positive definite. Hence the minimum occurs exactly when

$$\mathbf{v}_\ell - \mathbf{A}_\ell^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell = 0.$$

Therefore,

$$\mathbf{v}_\ell = (\mathbf{X}_\ell^H \mathbf{X}_\ell + \lambda \mathbf{I})^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell, \quad \ell = 1, 2. \quad (3.15)$$

The reduced biquaternion coefficient vector is then reconstructed from the two complex solutions as

$$\mathbf{v}_{\text{RB-QCR}} = \mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2. \quad (3.16)$$

3.7 Reduced Biquaternion Linear Regression (RB-QLR)

In contrast to RB-QCR, Reduced Biquaternion Linear Regression (RB-QLR) reconstructs the test sample using training samples from each class separately. For each class $k \in \{1, \dots, K\}$, define the class-specific dictionary

$$\mathbf{X}^{(k)} \in \mathbb{H}_c^{HWT \times N_k}.$$

For a given test vector $\mathbf{y} \in \mathbb{H}_c^{HWT}$, RB-QLR computes a coefficient vector

$$\mathbf{v}_{\text{RB-QLR}}^{(k)} \in \mathbb{H}_c^{N_k}$$

by solving

$$\mathbf{v}_{\text{RB-QLR}}^{(k)} = \arg \min_{\mathbf{v} \in \mathbb{H}_c^{N_k}} \left\| \mathbf{y} - \mathbf{X}^{(k)} \mathbf{v} \right\|_2^2 + \lambda \|\mathbf{v}\|_2^2, \quad \lambda \geq 0. \quad (\text{RB-QLR})$$

We show that this problem, like RB-QCR, decomposes into two independent complex ridge regression problems. Using the idempotent representation in (3.12), write

$$\mathbf{y} = \mathbf{y}_1 \mathbf{e}_1 + \mathbf{y}_2 \mathbf{e}_2, \quad \mathbf{X}^{(k)} = \mathbf{X}_1^{(k)} \mathbf{e}_1 + \mathbf{X}_2^{(k)} \mathbf{e}_2, \quad \mathbf{v} = \mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2,$$

where

$$\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{C}^{HWT}, \quad \mathbf{X}_1^{(k)}, \mathbf{X}_2^{(k)} \in \mathbb{C}^{HWT \times N_k}, \quad \mathbf{v}_1, \mathbf{v}_2 \in \mathbb{C}^{N_k}.$$

By the same argument used for RB-QCR, the product separates as

$$\mathbf{X}^{(k)} \mathbf{v} = (\mathbf{X}_1^{(k)} \mathbf{v}_1) \mathbf{e}_1 + (\mathbf{X}_2^{(k)} \mathbf{v}_2) \mathbf{e}_2,$$

and applying the norm identity (3.13) yields

$$\begin{aligned} \left\| \mathbf{y} - \mathbf{X}^{(k)} \mathbf{v} \right\|_2^2 + \lambda \|\mathbf{v}\|_2^2 &= \frac{1}{2} \left(\left\| \mathbf{y}_1 - \mathbf{X}_1^{(k)} \mathbf{v}_1 \right\|_2^2 + \lambda \|\mathbf{v}_1\|_2^2 \right) \\ &\quad + \frac{1}{2} \left(\left\| \mathbf{y}_2 - \mathbf{X}_2^{(k)} \mathbf{v}_2 \right\|_2^2 + \lambda \|\mathbf{v}_2\|_2^2 \right). \end{aligned}$$

Since $\mathbf{v}_1$ and $\mathbf{v}_2$ are independent, the problem separates into

$$\min_{\mathbf{v}_\ell \in \mathbb{C}^{N_k}} \left\| \mathbf{y}_\ell - \mathbf{X}_\ell^{(k)} \mathbf{v}_\ell \right\|_2^2 + \lambda \|\mathbf{v}_\ell\|_2^2, \quad \ell = 1, 2. \quad (3.17)$$

For $\lambda > 0$, the closed-form solution follows directly from the same argument presented in the RB-QCR Section 3.6.

$$\mathbf{v}_\ell^{(k)} = \left(\mathbf{X}_\ell^{(k)H} \mathbf{X}_\ell^{(k)} + \lambda \mathbf{I} \right)^{-1} \mathbf{X}_\ell^{(k)H} \mathbf{y}_\ell, \quad \ell = 1, 2. \quad (3.18)$$

The reduced biquaternion class coefficient vector is then reconstructed from the two complex solutions as

$$\mathbf{v}_{\text{RB-QLR}}^{(k)} = \mathbf{v}_1^{(k)} \mathbf{e}_1 + \mathbf{v}_2^{(k)} \mathbf{e}_2. \quad (3.19)$$

The class- k reconstruction of the test sample is

$$\hat{\mathbf{y}}^{(k)} = \mathbf{X}^{(k)} \mathbf{v}_{\text{RB-QLR}}^{(k)},$$

and, by the norm identity (3.13), the corresponding residual is

$$r_k = \frac{1}{2} \left(\|\mathbf{y}_1 - \mathbf{X}_1^{(k)} \mathbf{v}_1^{(k)}\|_2^2 + \|\mathbf{y}_2 - \mathbf{X}_2^{(k)} \mathbf{v}_2^{(k)}\|_2^2 \right). \quad (3.20)$$

Finally, the test sample $\mathbf{y}$ is assigned to the class with the smallest reconstruction residual,

$$k^* = \arg \min_{k \in \{1, \dots, K\}} r_k. \quad (3.21)$$

Algorithm 3 Reduced Biquaternion Linear Regression Classification (RB-QLR)

Input: Class-wise training tensors $\{\mathcal{X}_j^{(k)}\}_{j=1}^{N_k} \subset \mathbb{H}_c^{H \times W \times T}$, $k = 1, \dots, K$; a test tensor $\mathcal{Y} \in \mathbb{H}_c^{H \times W \times T}$; regularization parameter $\lambda \geq 0$.

Output: Predicted class label $\hat{k}$.

1. Vectorize: $\mathbf{x}_j^{(k)} = \text{Vec}(\mathcal{X}_j^{(k)})$, $\mathbf{y} = \text{Vec}(\mathcal{Y})$, and form $\mathbf{X}^{(k)} = [\mathbf{x}_1^{(k)}, \dots, \mathbf{x}_{N_k}^{(k)}]$, $k = 1, \dots, K$.
2. Normalize the columns of each $\mathbf{X}^{(k)}$.
3. Decompose $\mathbf{X}^{(k)} = \mathbf{X}_1^{(k)} \mathbf{e}_1 + \mathbf{X}_2^{(k)} \mathbf{e}_2$ and $\mathbf{y} = \mathbf{y}_1 \mathbf{e}_1 + \mathbf{y}_2 \mathbf{e}_2$.
4. For each class k and each $\ell = 1, 2$, compute

$$\mathbf{v}_\ell^{(k)} = (\mathbf{X}_\ell^{(k)H} \mathbf{X}_\ell^{(k)} + \lambda I)^{-1} \mathbf{X}_\ell^{(k)H} \mathbf{y}_\ell \quad (\text{for } \lambda > 0)$$

5. Reconstruct $\mathbf{v}_{\text{RB-QLR}}^{(k)} = \mathbf{v}_1^{(k)} \mathbf{e}_1 + \mathbf{v}_2^{(k)} \mathbf{e}_2$.
6. Compute the class residuals

$$r_k = \frac{1}{2} \left(\|\mathbf{y}_1 - \mathbf{X}_1^{(k)} \mathbf{v}_1^{(k)}\|_2^2 + \|\mathbf{y}_2 - \mathbf{X}_2^{(k)} \mathbf{v}_2^{(k)}\|_2^2 \right).$$

7. Classify: $\hat{k} = \arg \min_{k \in \{1, \dots, K\}} r_k$.
-

3.8 Classification via Collaborative Residuals

Let $\mathcal{I}_k$ denote the index set of training samples belonging to class k . Define the selection operator

$$\delta_k : \mathbb{H}_c^N \rightarrow \mathbb{H}_c^N, \quad \delta_k(\mathbf{v})_i = \begin{cases} v_i, & i \in \mathcal{I}_k, \\ 0, & i \notin \mathcal{I}_k. \end{cases}$$

Applying δ_k to the RB-QCR coefficient vector yields the class-specific coefficients

$$\mathbf{v}^{(k)} = \delta_k(\mathbf{v}_{\text{RB-QCR}}).$$

The class- k reconstruction of the test sample is then

$$\hat{\mathbf{y}}^{(k)} = \mathbf{X} \mathbf{v}^{(k)}.$$

Using the idempotent decomposition of the reduced biquaternion representation introduced in Section 2 and the norm identity in (3.13), the corresponding reconstruction residual becomes

$$r_k = \frac{1}{2} \left(\|\mathbf{y}_1 - \mathbf{X}_1 \mathbf{v}_1^{(k)}\|_2^2 + \|\mathbf{y}_2 - \mathbf{X}_2 \mathbf{v}_2^{(k)}\|_2^2 \right). \quad (3.22)$$

Finally, the test sample $\mathbf{y}$ is assigned to the class with the smallest reconstruction residual,

$$k^* = \arg \min_k r_k. \quad (3.23)$$

Algorithm 4 Reduced Biquaternion Collaborative Representation Classification (RB-QCR)

Input: Reduced biquaternion training tensors $\mathcal{X}_1, \dots, \mathcal{X}_N \in \mathbb{H}_c^{H \times W \times T}$ with class labels from K classes; a test tensor $\mathcal{Y} \in \mathbb{H}_c^{H \times W \times T}$; regularization parameter $\lambda > 0$.

Output: Predicted class label $\hat{k}$.

1. Vectorize: $\mathbf{x}_j = \text{Vec}(\mathcal{X}_j)$, $j = 1, \dots, N$, and $\mathbf{y} = \text{Vec}(\mathcal{Y})$; form the global matrix $\mathbf{X} = [\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(K)}]$.
2. Normalize the columns of $\mathbf{X}$ to unit ℓ_2 -norm.
3. Decompose $\mathbf{X} = \mathbf{X}_1 \mathbf{e}_1 + \mathbf{X}_2 \mathbf{e}_2$ and $\mathbf{y} = \mathbf{y}_1 \mathbf{e}_1 + \mathbf{y}_2 \mathbf{e}_2$.
4. Solve, for $\ell = 1, 2$,

$$\mathbf{v}_\ell = (\mathbf{X}_\ell^H \mathbf{X}_\ell + \lambda I)^{-1} \mathbf{X}_\ell^H \mathbf{y}_\ell.$$
5. Reconstruct $\mathbf{v}_{\text{RB-QCR}} = \mathbf{v}_1 \mathbf{e}_1 + \mathbf{v}_2 \mathbf{e}_2$.
6. For each class $k = 1, \dots, K$, extract $\mathbf{v}^{(k)} = \delta_k(\mathbf{v}_{\text{RB-QCR}})$ and compute the residual

$$r_k = \frac{1}{2} \left(\|\mathbf{y}_1 - \mathbf{X}_1 \mathbf{v}_1^{(k)}\|_2^2 + \|\mathbf{y}_2 - \mathbf{X}_2 \mathbf{v}_2^{(k)}\|_2^2 \right).$$

7. Classify: $\hat{k} = \arg \min_{k \in \{1, \dots, K\}} r_k$.
-

Chapter 4

Experimental Results

In this chapter, we assess the models developed in Chapter 3. The chapter is organized as follows. Section 4.1 describes the two benchmark datasets used throughout the experiments, RAVDESS and CREMA-D, and Section 4.2 details the pre-processing pipeline, including face detection, spatial downsampling, and uniform frame sampling. Section 4.3.1 then compares the four quaternion-structured regression classifiers on RAVDESS and CREMA-D under actor-independent splits, and Section 4.3.2 summarizes their stability by reporting the minimum, maximum, standard deviation, and mean test accuracy across the eight random seeds. Section 4.3.3 evaluates the robustness of RB-QCR when RAVDESS and CREMA-D clips are pooled and split under different train/test proportions, and Section 4.3.4 studies how accuracy degrades as the number of emotion classes increases from the binary happy-versus-sad task. The remaining sections analyze the RB-QCR model in more detail: Section 4.4.1 examines how accuracy changes with spatial resolution and the number of sampled frames, and Section 4.4.2 derives its time complexity, while Section 4.4.3 identifies the most important facial regions by occluding each patch and recording the resulting accuracy drop. Finally, Section 4.4.4 tests whether the reduced biquaternion representation retains enough identity information to recognize actors when the same actors appear in both train and test sets, and Section 4.4.5 measures how robust that actor-identification model is to fixed rotations and

random per-clip rotation jitter. Throughout the remainder of this chapter, QLR and QCR denote the Hamilton quaternion linear regression and collaborative representation classifiers, respectively, whereas RB-QLR and RB-QCR denote the corresponding reduced biquaternion classifiers.

Table 4.1 summarizes the analyses conducted in this chapter, the models involved, and the specific sections in which each analysis is presented.

Table 4.1: Summary of experimental analyses

Analysis	Models involved	Section
Classification results	RB-QCR, RB-QLR, QCR, QLR	4.3.1
Descriptive statistics	RB-QCR, RB-QLR, QCR, QLR	4.3.2
RAVDESS-CREMA-D mixed-dataset split	RB-QCR	4.3.3
Multi-emotion classification	RB-QCR, RB-QLR	4.3.4
Time complexity	RB-QCR	4.4.2
Frame-size analysis	RB-QCR	4.4.1
Face-region occlusion	RB-QCR	4.4.3
Facial recognition / actor identification	RB-QCR, RB-QLR	4.4.4
Actor-ID rotation analysis	RB-QCR	4.4.5

4.1 Datasets

All experiments in this thesis are conducted on two publicly available audio-visual emotion datasets: the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) and the Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D). Both datasets consist of short, labelled video recordings of actors delivering scripted speech while portraying specific emotional states.

Throughout the experimental analysis, the classification problem is restricted to two emotion categories: *happy* and *sad*, across all available emotion intensity levels from each dataset.

4.1.1 RAVDESS

The RAVDESS dataset contains 1,440 video clips from 24 professional actors, equally divided between male and female performers. In each recording, actors deliver short scripted English utterances while expressing a target emotion; Figure 4.1 shows sample frames taken from two such recordings, one from each emotion class. All recordings were captured in a controlled studio environment with a neutral background and consistent lighting conditions. Each actor performs the same utterances across multiple emotion categories and intensity levels. Under the binary happy-versus-sad restriction, RAVDESS contributes 384 video clips in total, with 192 clips per emotion category. Since each actor contributes an equal number of clips to both classes, the resulting dataset remains balanced at the actor level.



Figure 4.1: Sample frames from the RAVDESS dataset showing actors displaying happy (left) and sad (right) expressions.

4.1.2 CREMA-D

The CREMA-D dataset contains video clips from 91 actors performing scripted emotional speech. Similar to RAVDESS, only the *happy* and *sad* emotion categories were considered, resulting in 2,542 video clips, with 1,271 clips per emotion category; Figure 4.2 shows sample

frames from two such recordings, one from each emotion class. The dataset has greater variation in facial appearance, speaking style, expression intensity, and recording conditions. Unlike the controlled studio environment used in RAVDESS, CREMA-D recordings were captured under less constrained conditions using a green-screen background, resulting in a dataset that is generally less polished and more variable.

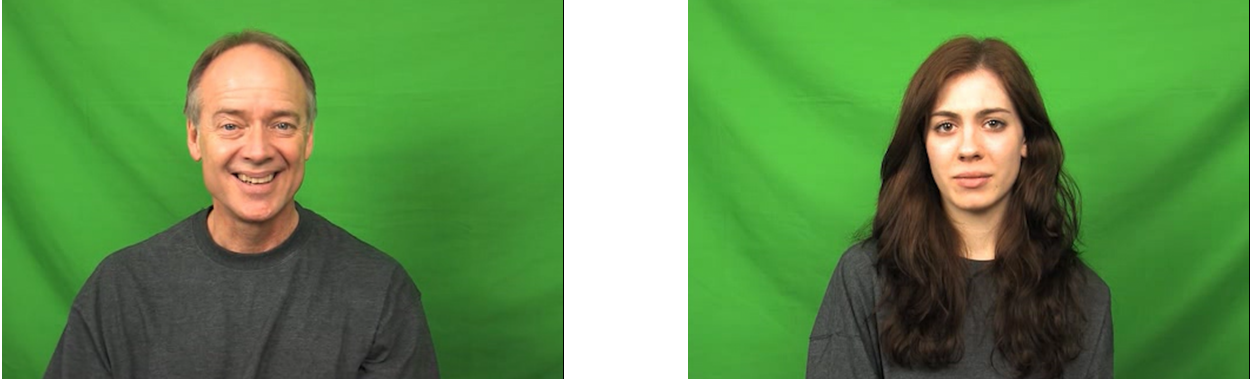


Figure 4.2: Sample frames from the CREMA-D dataset showing actors displaying happy (left) and sad (right) expressions.

4.2 Data Processing

To improve classification performance, we applied several pre-processing steps to the video data. First, all videos were trimmed to a fixed duration of three seconds to ensure consistency across samples. Next, a Haar cascade face detection algorithm was used to localize the facial region in each video and reduce background noise. Finally, for both datasets, the spatial resolution was downsampled to 64×64 pixels, and 25 frames were uniformly sampled at fixed temporal intervals. Multiple spatial and temporal configurations were evaluated to study the effect of changing input video dimensions on prediction accuracy; the detailed results of these experiments are presented in the Frame-size analysis,

4.3 Model Evaluation

4.3.1 Classification Results

The main experiments use actor-independent train, validation, and test splits. In this setting, the partition is performed at the actor level rather than at the individual video level. Therefore, all clips belonging to a given actor are assigned to only one subset: training, validation, or testing. No actor appearing in the training set appears in the validation or test set. This reduces identity leakage and provides a stricter evaluation of whether the classifiers generalize to unseen subjects rather than relying on actor-specific facial structure, appearance, or recording characteristics.

For each dataset, the actors are split into 80.00% training, 10.00% validation, and 10.00% testing subsets. The validation set is used only for selecting the regularization parameter in the collaborative representation classifiers. The candidate grid is

$$\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}.$$

After selecting λ , the model is evaluated on the held-out actor-disjoint test set. To check replication consistency, each experiment is repeated across eight random seeds.

A practical limitation was observed for the Hamilton quaternion models on CREMA-D. The quaternion implementation uses the real matrix representation $P(X)$, which expands the regression system from quaternion-valued data into a real-valued system with substantially larger dimensions. While this representation fit in memory for RAVDESS at the main preprocessing configuration of $64 \times 64 \times 25$, it did not fit in memory for CREMA-D at the same resolution because of the larger number of training clips. Therefore, the CREMA-D quaternion experiments were downscaled to $32 \times 32 \times 18$. This is an important practical distinction: the reduced biquaternion models are not only competitive in accuracy, but also require less memory and run faster because they avoid the full Hamilton quaternion $P(X)$

expansion.

Table 4.2 reports the best observed test accuracy from the eight random seeds for each model and dataset. Accuracy is defined as the proportion of correctly classified samples in the test set. These results show that RB-QCR is the strongest classifier among the reduced biquaternion and Hamilton quaternion models considered here. On RAVDESS, RB-QCR reaches 98.75% accuracy, exceeding the best QCR result of 97.50%. On CREMA-D, RB-QCR again achieves the highest observed accuracy, 96.63%, compared with 95.83% for QCR.

Table 4.2: Best observed test accuracy (%) across eight random seeds

Dataset	RB-QCR	RB-QLR	QCR	QLR
RAVDESS	98.75	88.75	97.50	88.75
CREMA-D	96.63	93.06	95.83	91.07

The confusion matrices in Tables 4.3 and 4.4 provide a detailed breakdown of the classification performance by comparing the predicted emotion labels against the ground-truth labels. The diagonal entries represent correctly classified samples for each emotion, while the off-diagonal entries indicate misclassifications.

The best RB-QCR result on RAVDESS is obtained with $\lambda = 10^{-1}$, producing the confusion matrix shown in Table 4.3. The classifier correctly identifies all 40 sad clips and 39 of the 40 happy clips, with only a single happy sample misclassified as sad. No sad samples are incorrectly predicted as happy.

The best RB-QCR result on CREMA-D is obtained with $\lambda = 10^{-2}$, producing the confusion matrix shown in Table 4.4. Of the 252 happy clips, 242 are correctly classified, while 10 are misclassified as sad. Similarly, 245 of the 252 sad clips are correctly identified, with only 7 predicted as happy.

The relatively small and balanced number of misclassifications across both classes indicates that the model effectively captures the underlying facial characteristics associated with each emotion. Moreover, the similar number of errors for the happy and sad classes suggests

that the classifier performs consistently without exhibiting a noticeable bias toward either emotion.

Table 4.3: Confusion matrix for the best RB-QCR run on RAVDESS (accuracy = 98.75%)

	Predicted Happy	Predicted Sad
Actual Happy	39	1
Actual Sad	0	40

Table 4.4: Confusion matrix for the best RB-QCR run on CREMA-D (accuracy = 96.63%)

	Predicted Happy	Predicted Sad
Actual Happy	242	10
Actual Sad	7	245

4.3.2 Descriptive Statistics

Table 4.5 reports the minimum, maximum, standard deviation, and mean test accuracy across the eight random seeds for the reduced biquaternion and Hamilton quaternion models on both datasets. The standard deviation is reported in percentage points and measures how sensitive each model is to the actor-independent train, validation, and test split.

Table 4.5: Descriptive statistics of test accuracy across eight random seeds

Dataset	Model	Min (%)	Max (%)	Std. Dev. (%)	Mean (%)
RAVDESS	RB-QCR	87.50	98.75	3.95	93.12
RAVDESS	RB-QLR	61.25	88.75	9.18	77.81
RAVDESS	QCR	87.50	97.50	3.76	92.81
RAVDESS	QLR	61.25	88.75	9.18	77.81
CREMA-D	RB-QCR	87.30	96.63	3.48	92.67
CREMA-D	RB-QLR	86.71	93.06	2.46	89.06
CREMA-D	QCR	87.50	95.83	3.31	92.34
CREMA-D	QLR	83.07	91.07	3.17	86.70

The statistics show two consistent patterns. First, collaborative representation is substantially stronger than linear regression within both quaternion-structured representation domains. This is most visible on RAVDESS, where RB-QCR has mean accuracy 93.12% compared with 77.81% for RB-QLR, and QCR has mean accuracy 92.81% compared with 77.81% for QLR. Second, RB-QCR has the highest mean accuracy on CREMA-D and the highest best observed accuracy on both datasets. The reduced biquaternion model therefore gives the best overall balance between accuracy, algebraic structure, and computational feasibility.

4.3.3 Out of Distribution Evaluation

The RAVDESS–CREMA-D mixed-dataset analysis evaluates how RB-QCR behaves when clips from both datasets are pooled into a single binary happy-versus-sad classification problem. This setting is different from the main per-dataset experiments. Instead of training and testing within one recording environment, the model is exposed to the combined variation of RAVDESS and CREMA-D, including differences in actors, recording conditions, expression style, video quality, and background structure.

The experiment uses a fixed regularization value $\lambda = 10^{-4}$. The purpose is to assess whether the reduced biquaternion collaborative representation classifier remains stable when the training and testing data are drawn from a more heterogeneous pooled dataset.

Table 4.6 summarizes the proportions of training and testing samples used in the cross-dataset experiments after combining the RAVDESS and CREMA-D datasets. For example, a 50/50 split indicates that 50.00% of the videos from each dataset were allocated to the training set, with the remaining 50.00% used for testing. All reported classification accuracies are computed on the test set.

Table 4.6: RAVDESS–CREMA-D mixed-dataset split analysis for RB-QCR at fixed $\lambda = 10^{-4}$

Train/test split	Train clips	Test clips	RB-QCR accuracy (%)
50/50	1463	1463	90.73
60/40	1756	1170	90.71
70/30	2048	878	90.80
80/20	2341	585	93.20
90/10	2633	293	93.35
Mean	–	–	91.76

The mixed-dataset results show that RB-QCR remains stable when RAVDESS and CREMA-D are combined. Across the five split settings, the accuracy ranges from 90.71% to 93.35%, with a mean accuracy of 91.76%. This indicates that the reduced biquaternion representation maintains strong classification performance even when the data contain greater variability in actor identity, recording environment, background structure, and expression style.

The strongest RB-QCR results occur when the training proportion is larger. At the 80/20 split, RB-QCR reaches 93.20%, and at the 90/10 split it reaches 93.35%. This suggests that the reduced biquaternion classifier benefits from the larger and more heterogeneous training set produced by pooling RAVDESS and CREMA-D. At the same time, the accuracy is lower than the best within-dataset results, which is expected because the classifier must handle both emotion-level variation and dataset-level variation simultaneously.

4.3.4 Extension from Binary to Multi-Emotion Classification

To evaluate how the proposed reduced biquaternion models behave as the classification problem becomes more difficult, an additional experiment was conducted by gradually increasing the number of emotion classes. Five emotion sets were considered. The three-class setting used happy, sad, and disgust. Angry was added in the four-class setting, fearful in the

five-class setting, calm in the six-class setting, and surprised in the seven-class setting.

Table 4.7 reports the mean classification accuracy over eight random seeds for both the reduced biquaternion linear regression classifier and the reduced biquaternion collaborative representation classifier evaluated on RAVDESS.

Table 4.7: Classification accuracy under increasing emotion-set size

Number of emotions	Emotion set	RB-QLR accuracy (%)	RB-QCR accuracy (%)
3	Happy, Sad, Disgust	58.44 ± 8.73	73.33 ± 3.33
4	+ Angry	47.81 ± 5.99	66.48 ± 2.54
5	+ Fearful	38.81 ± 4.08	58.94 ± 2.86
6	+ Calm	36.82 ± 4.74	56.15 ± 3.15
7	+ Surprised	32.05 ± 3.97	49.38 ± 3.01

As expected, accuracy decreases as more emotion categories are introduced. This is natural because the classification task becomes harder when the model must separate a larger number of affective classes, several of which have overlapping visual and temporal patterns. However, the decline is gradual rather than unstable. The collaborative representation model consistently outperforms the linear regression classifier across all emotion-set sizes. On average, RB-QCR outperformed RB-QLR by 18.00% across all emotion-set sizes.

The RB-QCR model decreases from 73.33% accuracy in the three-class setting to 49.38% in the seven-class setting, while remaining well above the seven-class chance level of approximately 14.29%.

This suggests that the collaborative residual-based decision rule is more robust than the direct linear regression classifier when the number of emotion categories increases. In particular, RB-QCR appears better able to use the joint representation of the test sample over the training dictionary, instead of relying only on a direct class-label regression fit.

4.4 Data Analysis

In the following analyses, we restrict our experiments to the RB-QCR model on the RAVDESS dataset due to the computational cost associated with training on two datasets. RB-QCR was selected because it achieved the highest classification accuracy among the evaluated models. Focusing on this model allows us to further assess its performance under different experimental conditions and gain a deeper understanding of its behaviour in different scenarios.

4.4.1 Frame Size Analysis

The frame-size analysis evaluates RB-QCR across different spatial resolutions and temporal sampling lengths. The number of sampled frames is varied over 5, 10, 15, 20, 25, and 30, while the spatial size is varied over 8, 16, 24, 32, 48, and 64 pixels. For each configuration, the reduced biquaternion feature dimension is $d = H \times W \times T$, where $H = W$ is the spatial size and T is the number of sampled frames.

Table 4.8 reports the classification accuracy of RB-QCR under different combinations of spatial resolution and temporal sampling length. The rows correspond to the spatial resolution of each video frame, while the columns correspond to the number of sampled frames used to represent each video sequence.

Table 4.8: RB-QCR accuracy (%) under different frame-size configurations

Spatial size	$T = 5$	$T = 10$	$T = 15$	$T = 20$	$T = 25$	$T = 30$
8×8	91.25	90.00	91.25	87.50	88.75	90.00
16×16	96.25	93.75	95.00	92.50	95.00	95.00
24×24	97.50	96.25	95.00	95.00	95.00	95.00
32×32	98.75	96.25	98.75	98.75	98.75	98.75
48×48	98.75	97.50	98.75	98.75	98.75	98.75
64×64	98.75	97.50	98.75	98.75	98.75	98.75

According to Table 4.8 the results show that spatial resolution has the strongest effect on performance. At 8×8 , the classifier loses important facial information and accuracy stays between 87.50% and 91.25%. Accuracy improves substantially once the spatial size increases to 16×16 and 24×24 , then largely saturates from 32×32 onward. Several configurations at 32×32 , 48×48 , and 64×64 reach 98.75% accuracy. These results indicate that the computational cost can be reduced substantially without a significant loss in classification accuracy. In particular, using moderate spatial resolutions and fewer sampled frames yields performance comparable to that of the largest frame sizes and longest temporal sequences. This suggests that RB-QCR does not require maximum spatial resolution or dense temporal sampling to achieve high recognition accuracy, enabling a more computationally efficient implementation with reduced memory requirements and faster processing.

The temporal dimension has a weaker and less monotone effect. Increasing the number of sampled frames does not consistently improve accuracy at a fixed spatial resolution. For example, at 64×64 , the accuracy is already 98.75% with 5 frames and remains at 98.75% for 15 through 30 frames, except for 10 frames where it is 97.50%. This suggests that, for the happy-versus-sad setting, the cropped facial expression structure carries more discriminative signal than dense temporal sampling. The main experiments use $64 \times 64 \times 25$ because it preserves detailed facial structure while remaining computationally feasible for the reduced biquaternion model.

4.4.2 Time Complexity

In this section we analyze the time complexity of Algorithm 4 in Section 3.8 for RB-QCR. Table 4.9 reports the cost of each step of the algorithm. We measure the cost of each step using big- O notation, which records how the number of arithmetic operations grows as $d = HWT$ and N increase, ignoring constant factors. The counts in Table 4.9 are obtained as follows. Steps 1 and 2 process each of the dN entries of the training data a fixed number of times, so their cost is $O(dN)$. In Step 3, the Gram matrix $\mathbf{X}_\ell^H \mathbf{X}_\ell$ has N^2 entries, and

each entry is an inner product of two length- d columns requiring d multiplications, giving $O(dN^2)$; the vector $\mathbf{X}_\ell^H \mathbf{y}_\ell$ consists of N such inner products, giving $O(dN)$; and solving the resulting $N \times N$ linear system by a Cholesky factorization costs $O(N^3)$, a standard result in numerical linear algebra. Step 5 computes one reconstruction per class using only that class's N_k columns, and since $\sum_k N_k = N$, the total cost over all classes is $O(dN)$. Lastly, in step 6, since we calculated K residuals in step 5, we select the smallest residual among them. This requires scanning all k residual values once, resulting in a computational complexity of $O(K)$.

Table 4.9: Computational complexity of Algorithm 4 (RB-QCR).

Step	Operation	Complexity
1	Vectorization of training and test tensors	$O(dN)$
2	Idempotent decomposition X_ℓ, y_ℓ	$O(dN)$
3a	Gram matrices $X_\ell^H X_\ell$	$O(dN^2)$
3b	Right-hand sides $X_\ell^H y_\ell$	$O(dN)$
3c	Cholesky solve of $(X_\ell^H X_\ell + \lambda I)v_\ell = X_\ell^H y_\ell$	$O(N^3)$
4	Reconstruction $v = v_1 e_1 + v_2 e_2$	$O(N)$
5	Class residuals $r_k(\mathbf{y})$, $k = 1, \dots, K$	$O(dN)$
6	Classification $\arg \min_k r_k(\mathbf{y})$	$O(K)$
Total		$O(dN^2 + N^3)$

The overall cost is therefore $O(dN^2 + N^3)$, dominated by Step 3. Note, since N is the number of training videos and is comparatively small ($d \gg N$). Consequently, the $O(N^3)$ term is negligible relative to the formation of the Gram matrices, and the effective complexity of RB-QCR is $O(dN^2)$.

4.4.3 Face-Region Occlusion Analysis

The occlusion analysis evaluates the robustness of RB-QCR by measuring the effect of localized pixel-level perturbations on classification performance. To do so, the face region is divided into an 8×8 grid of patches. Each patch is individually occluded across all sampled frames, and the model is re-evaluated to measure the resulting change in performance.

To quantify this change, let $\text{Acc}_{\text{baseline}}$ denote the baseline accuracy of RB-QCR without

any occluded patches applied to the input videos, and let $\text{Acc}_{\text{occluded}(i,j)}$ denote the classification accuracy after introducing an occluding patch at location (i,j) . The importance of each patch is then measured by the change in accuracy:

$$\Delta_{ij} = \text{Acc}_{\text{baseline}} - \text{Acc}_{\text{occluded}(i,j)}.$$

The baseline accuracy for this analysis is 98.75%. Larger positive values of Δ_{ij} indicate that the occluded patch is more important for classification.

Figure 4.3 illustrates the spatial distribution of the occlusion sensitivity across the facial regions. Each cell corresponds to a facial patch, and its colour indicates the reduction in classification accuracy when that patch is occluded. Warmer colours (red) represent larger accuracy drops, indicating facial regions that are more influential in the model’s emotion classification decisions.

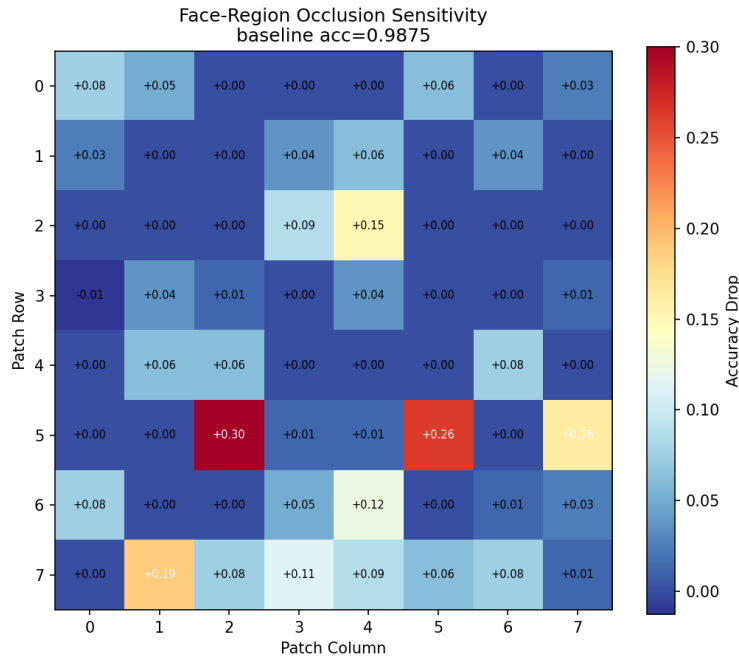


Figure 4.3: Face-region occlusion sensitivity heatmap for RB-QCR. Each cell gives the accuracy drop after occluding the corresponding facial patch across all frames.

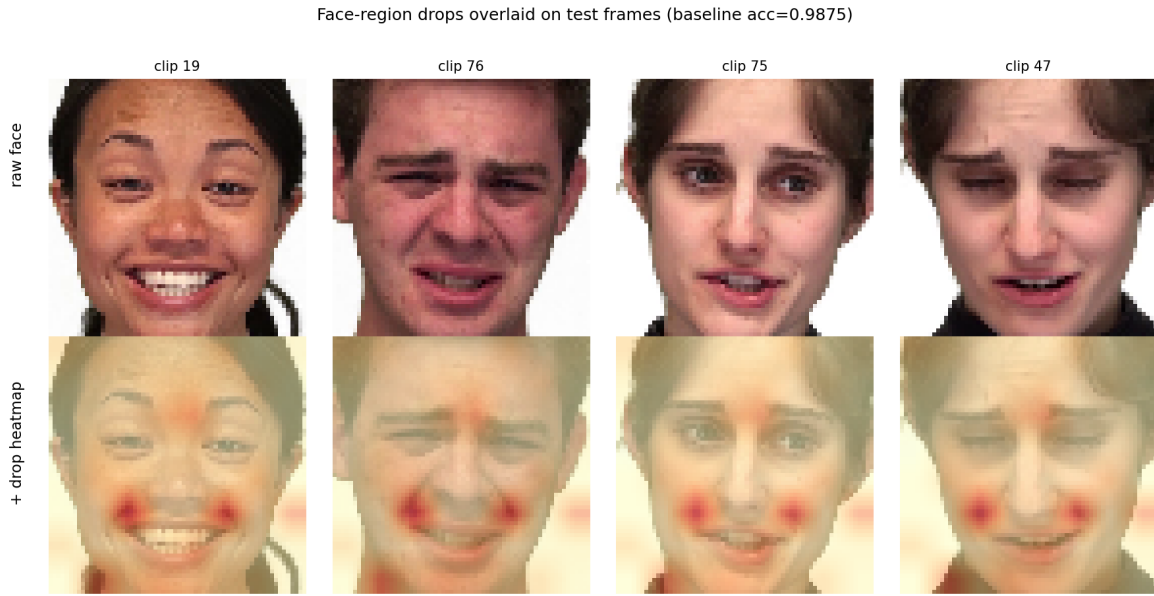


Figure 4.4: Occlusion sensitivity overlaid on sample test frames. The strongest drops occur around the mouth, cheeks, and lower facial regions.

Figure 4.4 shows the occlusion sensitivity heatmap overlaid on the facial structure of the actors. According to Figures 4.3 and 4.4 the largest drops occur at patch (5, 2) with an accuracy decrease of 30.00%, patch (5, 5) with a decrease of 26.25%, patch (7, 1) with a decrease of 18.75%, and patch (5, 7) with a decrease of 16.25%. These patches correspond primarily to the lower face, including the mouth corners, cheeks, and chin/lower-face boundary. This is consistent with the happy-versus-sad task, where mouth curvature and cheek movement are highly informative. Some upper-face and side regions also contribute, such as patch (2, 4) with a drop of 15.00% and patch (6, 4) with a drop of 12.50%. However, many patches have zero or nearly zero drop, and one patch has a small negative value. A negative value means that occluding that patch slightly improved accuracy on the test set, which can occur when the region contains nuisance variation or background artifacts. Overall, the occlusion analysis indicates that RB-QCR is not using the entire face uniformly. Its classification decisions are most sensitive to expression-bearing regions around the mouth and lower face.

4.4.4 Facial Recognition and Actor Identification Experiment

A separate actor-identification experiment was conducted as a diagnostic study of the RB-QCR model to assess its ability to learn actor-specific facial structures and perform in face recognition. In the main experiments, actor-independent splits are used, so the actors appearing in the test set are not present in the training set. This prevents direct identity memorization across the train and test sets. In the actor-identification experiment, the class label is changed from the emotion label to the actor ID. Unlike the main emotion-classification experiments, this diagnostic uses a within-actor clip split, so each actor appears in both the training and test sets. This design intentionally allows identity information to be shared across the split.

Table 4.10 reports the actor-identification performance of the RB-QCR. The table includes the number of actors considered, the number of training and testing clips used in the experiment, and the resulting mean identification accuracy for each model across multiple seeds.

Table 4.10: Actor-identification accuracy using the reduced biquaternion representation

Model	Number of actors	Train clips	Test clips	Mean accuracy (%)
RB-QLR actor ID	24	312	72	99.31
RB-QCR actor ID	24	312	72	100.00

The RB-QCR actor-identification model achieves 100.00% mean accuracy across the eight seeds. This shows that the reduced biquaternion representation preserves strong identity-specific facial information when the same actors are present in both train and test sets. The result reinforces the need for actor-independent splitting in the main emotion experiments. Since the main results use actor-disjoint train, validation, and test sets, the classifier cannot obtain high test accuracy simply by memorizing actor identities from the training set. The actor-identification experiment therefore functions as a leakage diagnostic: it shows what would be possible under a weaker split, and it justifies the stricter actor-independent protocol

used for the reported emotion-classification results.

4.4.5 Actor Identification with Rotation Analysis

The actor-ID rotation analysis evaluates whether the actor-identification model discussed in section 4.4.4 is robust to the geometric perturbations introduced through image rotations. Two perturbation settings are considered. In the first, every test clip is rotated by a fixed angle. In the second, each test clip is randomly rotated within a jitter range $[-A, A]$, and the experiment is repeated over five random jitter seeds.

Figure 4.5 presents the actor-identification accuracy under two types of rotation perturbations, fixed-angle rotations and random per-clip rotation jitter. In the fixed-angle sweep, the accuracy degradation is not symmetric, positive rotations are tolerated somewhat better than negative ones, most notably at $\pm 10^\circ$ and $\pm 15^\circ$, where accuracy remains near 83.00% for $+10^\circ$ but falls to approximately 74.00% for -10° while at $\pm 15^\circ$, accuracy is 41.67% for $+15^\circ$ compared with 23.61% for -15° which is almost double the classification accuracy. In the random jitter experiment, a clear trend emerges, as the jitter range increases, the mean accuracy decreases while the standard deviation across runs grows, indicating that larger perturbations not only degrade performance but also make it increasingly unstable.

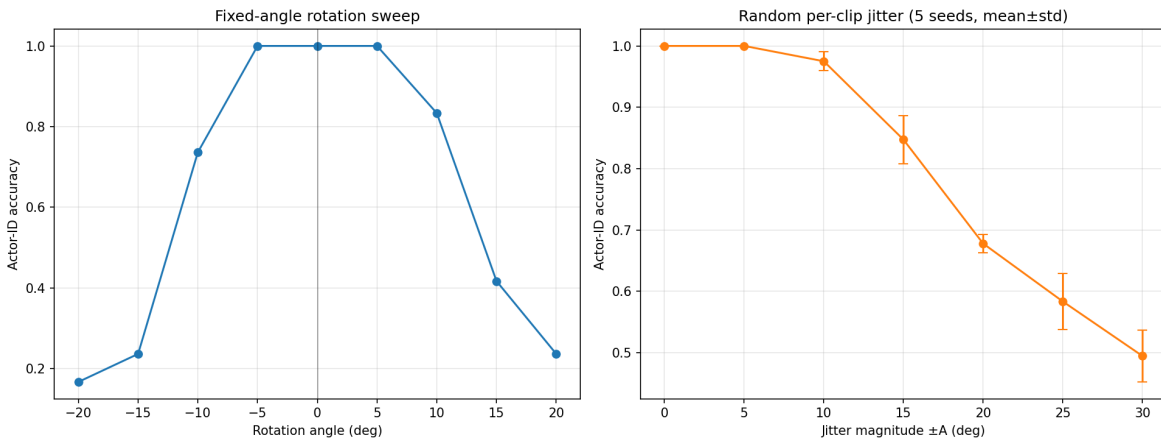


Figure 4.5: Actor-ID accuracy under fixed rotations and random per-clip rotation jitter.

Table 4.11 summarizes the results of the random per-clip rotation jitter experiment.

For each jitter magnitude, the table reports the mean actor-identification accuracy. The random-jitter results are smoother than the fixed-angle sweep. Accuracy remains essentially unchanged for $A = 5^\circ$ and remains high at $A = 10^\circ$, with mean accuracy 97.50%. After that point, performance decreases steadily: 84.72% at $A = 15^\circ$, 67.78% at $A = 20^\circ$, and 49.44% at $A = 30^\circ$. The rotation experiment therefore shows that the reduced biquaternion actor-ID model is robust to small alignment changes but not invariant to large rotations.

Table 4.11: Random per-clip rotation jitter accuracy for actor identification

Jitter magnitude A	Mean accuracy (%)	Standard deviation (%)
0°	100.00	0.00
5°	100.00	0.00
10°	97.50	1.58
15°	84.72	3.98
20°	67.78	1.50
25°	58.33	4.42
30°	49.44	3.84

Table 4.12 reports the actor-identification accuracy for each fixed rotation angle. Each row corresponds to a specific rotation applied uniformly to all test clips. For fixed rotations, accuracy remains high near the training orientation. At 0° , accuracy is 100.00%, and at $+5^\circ$ it reaches 100.00%. At -5° , accuracy remains 100.00%. However, performance drops sharply as the rotation becomes larger. Accuracy falls to 73.61% at -10° , 41.67% at $+15^\circ$, and 16.67% at -20° . This shows that the actor-identification model is sensitive to pose and image alignment.

Table 4.12: Fixed-angle rotation accuracy for actor identification

Rotation angle	Accuracy (%)
-20°	16.67
-15°	23.61
-10°	73.61
-5°	100.00
0°	100.00
$+5^\circ$	100.00
$+10^\circ$	83.33
$+15^\circ$	41.67
$+20^\circ$	23.61

Figure 4.6 illustrates examples of the facial crops after applying different rotations.



Figure 4.6: Example test-time rotations applied to a face crop. Larger rotations introduce substantial changes in alignment and visible facial geometry.

Chapter 5

Conclusion

This thesis examined quaternion and reduced biquaternion regression models for video emotion recognition. The central question was whether colour video clips could be represented in an algebraically structured form and classified using regression-based methods, without depending on large deep learning architectures. The experiments focused on binary happy-versus-sad classification using the RAVDESS and CREMA-D datasets.

The quaternion models extended quaternion linear regression and quaternion collaborative representation from static colour images to video data. Each clip was encoded as a third-order quaternion tensor, with the RGB values of each pixel stored in the imaginary components of a quaternion. After vectorization, classification was performed by comparing reconstruction residuals, following the same principle used in linear regression classification and collaborative representation.

The reduced biquaternion models were developed to address the computational cost of the Hamilton quaternion formulation. The main algebraic advantage of the reduced biquaternion setting is its idempotent decomposition, which separates each sample into two complex-valued components. This allowed RB-QLR and RB-QCR to be solved through complex-valued linear systems instead of relying on the larger real-valued Hamilton quaternion embedding. In this way, the reduced biquaternion models kept a structured colour

representation while lowering the computational burden of the regression problem.

Among the models considered, RB-QCR gave the strongest overall results. Under actor-independent train, validation, and test splits, it achieved the highest observed accuracy on both RAVDESS and CREMA-D. This evaluation setting is important because the test actors are not seen during training, so the results reflect generalization to new subjects rather than memorization of actor-specific appearance. The mixed RAVDESS–CREMA-D experiment also indicated that RB-QCR remained stable when the classification problem included clips from both datasets.

The computational comparison was equally important. Hamilton quaternion QCR requires the expanded matrix representation $P(\mathbf{X}) \in \mathbb{R}^{4d \times 4n}$. At the $64 \times 64 \times 25$ setting, this representation became too large for the full CREMA-D experiment on a machine with 32 GB of RAM. The Hamilton quaternion CREMA-D experiments therefore had to be reduced to $32 \times 32 \times 18$. RB-QCR avoided this full expansion, which made it possible to retain the higher-resolution setting. This result shows that the reduced biquaternion formulation was not only competitive in accuracy, but also more practical in memory-limited settings.

The additional experiments gave a clearer picture of how RB-QCR behaves. The frame-size analysis showed that very small spatial resolutions remove useful facial information, while performance became more stable once the spatial resolution reached a moderate size. The face-region analysis suggested that the model relied mainly on meaningful facial regions rather than background content.

Taken together, the results support RB-QCR as the most practical model studied in this thesis. It produced the strongest classification performance among the quaternion-structured models, avoided the largest memory limitation of Hamilton quaternion QCR, and retained an interpretable idempotent-channel structure. These properties make reduced biquaternion collaborative representation a useful alternative for video emotion recognition when the available dataset is limited or when computational resources are constrained.

Two further directions remain open. First, the current implementation vectorizes each

quaternion-valued or reduced-biquaternion-valued video tensor before solving the regression problem. A natural next step is to develop a fully tensor-based formulation that operates directly on the multi-way video representation, rather than converting each clip into a long vector. Second, the present RB-QCR model is linear in its reconstruction step, and linear residuals can struggle to separate visually similar expressions such as anger and disgust or fear and surprise. A nonlinear extension for example, a kernel-based formulation of reduced biquaternion collaborative representation would allow the model to capture more subtle differences between closely related emotion classes and could improve discrimination on the most confusable emotion pairs in the multi-class setting.

Bibliography

- [1] E. Grassucci, D. Comminiello, and A. Uncini, “An information-theoretic perspective on proper quaternion variational autoencoders,” *Entropy*, vol. 23, no. 7, art. 856, 2021.
- [2] W. R. Hamilton, “On a new species of imaginary quantities connected with a theory of quaternions,” *Proc. Roy. Irish Acad.*, vol. 2, pp. 424–434, 1843.
- [3] F. L. Hitchcock, “The expression of a tensor or a polyadic as a sum of products,” *J. Math. Phys.*, vol. 6, no. 1–4, pp. 164–189, 1927.
- [4] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
- [5] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
- [6] B. Lee, H. Shin, B. Ku, and H. Ko, “Frame level emotion guided dynamic facial expression recognition with emotion grouping,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Vancouver, Canada, 2023, pp. 5681–5691.
- [7] J. Miao, K. I. Kou, and W. Liu, “Low-rank quaternion tensor completion for recovering color video and images,” *Pattern Recognit.*, vol. 107, art. 107505, 2020.
- [8] I. Naseem, R. Togneri, and M. Bennamoun, “Linear regression for face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 2106–2112, 2010.

- [9] S.-C. Pei, J.-H. Chang, and J.-J. Ding, “Commutative reduced biquaternions and their Fourier transform for signal and image processing applications,” *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 2012–2031, 2004.
- [10] S.-C. Pei, J.-H. Chang, J.-J. Ding, and M.-Y. Chen, “Eigenvalues and singular value decompositions of reduced biquaternion matrices,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 9, pp. 2673–2685, 2008.
- [11] L. Rodman, *Topics in Quaternion Linear Algebra*. Princeton, NJ, USA: Princeton Univ. Press, 2014.
- [12] E. Ryumina and A. Karpov, “Facial expression recognition using distance importance scores between facial landmarks,” in *Proc. 30th Int. Conf. Comput. Graph. Mach. Vis. (GraphiCon)*, vol. 2744, CEUR Workshop Proceedings, 2020.
- [13] S. J. Sangwine, “Fourier transforms of colour images using quaternion, or hypercomplex, numbers,” *Electron. Lett.*, vol. 32, no. 21, pp. 1979–1980, 1996.
- [14] J. Selva, A. S. Johansen, S. Escalera, K. Nasrollahi, T. B. Moeslund, and A. Clapés, “Video transformers: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12922–12943, 2023.
- [15] J. Sotelo-Barrales, D. Mata-Mendoza, M. Nakano-Miyatake, H. Pérez-Meana, and E. Escamilla-Hernández, “Dynamic facial expression recognition using geometric and deep features,” in *New Trends in Intelligent Software Methodologies, Tools and Techniques (SoMeT 2025)*. Amsterdam, The Netherlands: IOS Press, 2025, pp. 355–368.
- [16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, 2009.

- [17] C.-E. Yu, X. Liu, H. Luo, and Y. Zhang, “Reduced biquaternion tensors and applications in color video processing,” *J. Sci. Comput.*, vol. 106, no. 1, art. 7, 2026.
- [18] C. Yuan, X. Liu, Y. Yu, and Y. Zhang, “A direct method for solving regularized quaternion matrix regression problems,” *Math. Methods Appl. Sci.*, vol. 48, no. 11, pp. 11336–11343, 2025.
- [19] F. Zhang, M. Wei, Y. Li, and J. Zhao, “Special least squares solutions of the quaternion matrix equation $AX = B$ with applications,” *Appl. Math. Comput.*, vol. 270, pp. 425–433, 2015.
- [20] L. Zhang, M. Yang, and X. Feng, “Sparse representation or collaborative representation: Which helps face recognition?,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2011, pp. 471–478.
- [21] H. Zhou, L. Li, and H. Zhu, “Tensor regression with applications in neuroimaging data analysis,” *J. Amer. Statist. Assoc.*, vol. 108, no. 502, pp. 540–552, 2013.
- [22] X. Zhu, Y. Xu, H. Xu, and C. Chen, “Quaternion convolutional neural networks,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 631–647.
- [23] C. Zou, K. I. Kou, and Y. Wang, “Quaternion collaborative and sparse representation with application to color face recognition,” *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3287–3302, 2016.
- [24] C. Zou, K. I. Kou, L. Dong, X. Zheng, and Y. Y. Tang, “From greyscale to color: Quaternion linear regression for color face recognition,” *IEEE Access*, vol. 7, pp. 154131–154140, 2019.