
Proceedings Paper:
McKee, D, Townend, P, Webster, D et al. (2014) M-VCR : Multi-View Consensus Recognition for Real-Time Experimentation. In: IEEE International Symposium on Object/Component/Service-oriented Real-Time Distributed Computing. IEEE International Symposium on Object/Component/Service-oriented Real-Time Distributed Computing, 08-12 Jun 2014, Reno, NV, USA. IEEE, 76 - 83.

https://doi.org/10.1109/ISORC.2014.37

M-VCR: Multi-View Consensus Recognition for Real-Time Experimentation

David McKee, Paul Townend, David Webster, Jie Xu
School of Computing
University of Leeds
Leeds, UK
{scdwm, P.M.Townend, D.E.Webster, J.Xu}@leeds.ac.uk

Abstract— **A major application area in the computer vision domain is gesture recognition, requiring real-time image classification to respond to human interactions. However, current state-of-the-art high-quality algorithms for image classification do not meet many dynamic real-time requirements. This paper presents the development of M-VCR - a novel approach for improving the reliability of real-time image classification. M-VCR increases the quality of classifications under real-time constraints through the adoption of fast classification algorithms; although these algorithms individually produce lower quality results, utilisation under a 'consensus' approach can achieve results equivalent to those of much higher-quality algorithms. The proposed approach also allows for different algorithms to be utilised in parallel, building on the fault tolerance technique of N-versioning. A significant improvement in image classification is experimentally demonstrated for both the SURF and MSER feature detectors through our integration consensus approach. This improvement is delivered entirely through the integration method without requiring modification of the source algorithms being used.**

Keywords— Reliability, Gesture Recognition, Consensus, Real-Time, N-Versioning

I. INTRODUCTION

Since the early 1980s, image classification and specifically gesture recognition have been a significant part of computer vision research [1]. A large portion of the work has been on improving techniques for image segmentation and motion path recognition as well as object tracking. Gesture recognition can be defined as recognition of human gestures which are captured via a camera for the purposes of mapping particular gestures onto particular tasks. This typically consists of two stages:

1. Object detection and classification
2. Motion tracking

Given any individual video frame, the hand postures must be classified via image segmentation and feature detection. Image segmentation refers to the process of identifying the target object and removing the background from the image. A feature detector is then used to identify key points in the image typically based on colour gradients. In the realm of gesture recognition, given the target objects are hands, image segmentation techniques are commonly based on skin colour [2] and feature detection attempts to identify points such as finger tips. These features are then matched with a dataset of training images using a feature matching algorithm to provide a hand posture classification. The choice of this training set is often regarded as one of the most significant design decisions in computer vision systems. With subsequent video frames, the detected feature points are matched with each other, to provide a motion path for each point [3]. The combination of the motion paths and the individual image classifications at each frame form the basis for an overall classification of the gesture.

The purpose of gesture recognition is to enable natural interaction with computer interfaces for purposes of either communication or system control. In either situation there is a clear need for real-time computation. In particular when used for communication, common sign language requirements require a minimum of 4*fps* [4]. Therefore the entire process including image segmentation, feature detection and matching, as well as motion tracking must be able to consistently complete in at least 0.25*s*.

As a result, some work has been done by the likes of Kumarage et al. [5] and others [6]–[10] on "real-time" gesture recognition. It is noted however that with the exception of [5] and [8] the authors do not discuss the concept of timeliness. Rather, their focus is on improving the accuracy of classification and in the case of [3] and [7] they focus on tracking the motion paths. Although [5] and particularly [8] do discuss the timeliness of their approach it is again noticeable that their focus is on classifying motion paths rather than individual images. The computational complexity of calculating motion paths is at least an order of magnitude smaller than that for image classification. According to [8] motion tracking can be performed in 10s of milliseconds; as will be shown in section III this is not the case for image classification.

A key limitation that will be shown experimentally is that the speed of an algorithm appears to be inversely proportional to the quality of the results. Therefore in any real system there must be a trade-off between the timeliness of the results and the accuracy with which they classify images. This trade-off problem is similar to that identified by Avizienis et al. [11]–[13] and Littlewood [14] in the realm of dependability between: reliability, speed, and security. The Multi-View Consensus Recognition (M-VCR) framework proposed in this paper is derived from a trade-off between the reliability of results and the speed of obtaining them. This paper therefore considers some of the approaches that have been used in fault-tolerance to increase the dependability of systems given real-time constraints that have informed the development of M-VCR.

This work uses a traditional approach for gesture recognition in terms of its system architecture and then proposes a new architecture inspired by N-versioning [15], [16]. The proposed framework is then benchmarked against previous state-of-the-art and the results demonstrated in section VI show an improvement of 47% in classification whilst meeting the original timeliness constraints. It is vital to note that all the results demonstrated in this paper are entirely through the integration approach of M-VCR that utilises quantum super-positioning to achieve consensus and no modification of the classification algorithms has been attempted. The significance of treating the algorithms as commercial-off-the-shelf (COTS) tools allows us to evaluate the mathematical framework at an abstraction away from the problem of gesture recognition. It is therefore expected that the M-VCR framework could be applied to other scientific and engineering domains.

II. Related Work

A. Image Feature Matching Algorithms

In this subsection we briefly outline some of the state-of-the-art approaches to image matching as background for this work. Considered are therefore the following algorithms for feature detection: MSER, SURF, SIFT and ASIFT.

MSER, or the Maximally Stable Extremal Regions [17] algorithm is a region detector unlike the other detectors being considered. This algorithm focusses on the number of pixels in a 'blob' as well as the intensity of these pixels. The connectivity between 'blobs' is considered and in the process of matching the algorithm will merge 'blobs' until 'maximally stable' ones are found relating to the threshold of change between connected ones. Although this algorithm executes in worst-case O(n), relating to the number of pixels, as will be seen in section III the classification success rate is significantly lower than other approaches.

SIFT, or the Scale-Invariant Feature Transform suggested by Lowe [18] in 2004 was regarded as a significant breakthrough in terms of feature-detection and matching. During execution the algorithm detects key points in an image with a 128-value descriptor which is generated from histograms of key point orientations. The orientations are calculated from colour gradients and therefore not affected by either rotation or scale which had previously been significant barriers in the area of image feature matching. The problems of scale and rotation are particularly prevalent in the domain of gesture recognition. A greatly simplified representation of this algorithm's time complexity is $O(n^2 + n^2 \log n)$.

The SURF algorithm, otherwise known as Speeded Up Robust Feastures [19] was inspired by SIFT and utilises approximations in a similar fashion to MSER. It detects 'blobs' in the image and a descriptor is generated similar to that used in SIFT. Accordingly, due to less computation required the SURF algorithm outperforms SIFT in terms of speed however as will be seen later it doesn't match the quality of results achieved by SIFT. SURF can be regarded as being approximately $O(n^2)$ algorithm. In practice however it performs similar to MSER.

And finally the ASIFT, or Affine-SIFT algorithm proposed by Morel and Yu [20] in 2011 builds on SIFT to achieve fully affine invariant feature matching. The SIFT approach captures all but two of the affine parameters: translation, rotation, and zoom. ASIFT by using geometric mapping mathematically simulates the additional parameters (longitudinal and latitudinal angles of view) with a 'transition tilt'. It subsequently utilises SIFT to perform matching based on the remaining parameters. This approach demonstrates particularly high feature matching success rates, however it is also very slow (in excess of 1s per image classification compared to 0.01s) and therefore in its current form unsuitable for real-time gesture recognition.

For matching feature points or regions using either the SURF or MSER algorithms the normalised cross-correlation is used as defined by [21]. However for both the SIFT and ASIFT algorithms the descriptors are matched based on the rotation distance between them with the descriptors treated as vectors. This approach matches all points whose distances are below a threshold value. For the purposes of this work the default threshold parameters are used and no modifications to the algorithms are performed. Due to the temporal characteristics of these algorithms, the objective of the M-VCR framework is to allow the use of either SURF or MSER rather than SIFT or ASIFT.

B. From Fault Tolerance and Dependability

As previously mentioned M-VCR was inspired by approaches to fault tolerance, namely N-versioning [15], [16] as such a brief overview of N-versioning and its main alternative are considered here.

The principle idea for both N-versioning and Recovery Blocks [22] is that independent efforts to achieve the same result deliver greater confidence in the result. If these efforts are also different in method the confidence is even greater. In N-versioning this idea is extended to suggest concurrent efforts in achieving this result. This is extended even further with the concept of N-copy [23] where multiple instances of the input data occur and are then mapped to a single output.

Gesture recognition has a significant problem of timeliness. In particular the problem occurs during the process of image classification. This is due to the trade-off between the speed and quality of the feature detection and matching algorithms. It is demonstrated in this paper that the 'fast-enough' algorithms do not provide 'good-enough' image classifications for gesture recognition. As a result the proposed framework builds on N-versioning along with N-copy, rather than Recovery Blocks. As stated in [24] recovery blocks are not as appropriate for real-time fault tolerance since the recovery blocks require additional processing time to achieve a result. In the case of gesture recognition this would require the algorithms to be sequentially executed rather than run in parallel. As a result either the size of the training set or the frame rate which could be achieved would be reduced proportionally to the number of versions.

Fig. 1. ASL hand posture for the letter 'E', is there a gap between the palm of the hand and the thumb in the first image?

The remainder of this paper is structured as follows: in section IV the architecture of the proposed system is discussed. In section V the theoretical methodology of M-VCR is elaborated upon and then in the following section experimental results are portrayed. However, first in section III a summary of the scenario for which M-VCR was designed is provided.

III. MOTIVATING SCENARIO

In this section an example scenario is depicted and the key problem areas in gesture recognition which M-VCR aims to address are identified.

A. The Problem of Self Occlusion

A common problem in object recognition is that of self-occlusion, where part of an object hides the remainder. In the case of gestures this problem is commonplace with different parts of the hand hiding other parts consistently. This problem is best understood with the example of the hand posture in Figure 1, where due to self-occlusion by the fingers we cannot tell from the image at this angle whether they are touching the palm of the hand. This problem is addressed by [5] where multiple cameras at different angles of view are used. This technique of utilising multiple fields of view is not uncommon and in many cases more than two cameras are utilised in order to maintain a non-occluded view of the target object. This approach, when achieved with fully parallelised processing, does not incur a significant temporal impact.

For the purposes of generalising the use case for M-VCR it is worth noting that this problem of self-occlusion is not limited to the computer vision domain. Rather, it is common in most scientific experimenting and problem solving, but would probably be referred to as: "approaching the problem from a different angle". As a result M-VCR can be considered as integration approach for combining multiple solutions to the same problem from different perspectives. To the best of the authors' knowledge there has been no previous attempt to automatically combine experimental data in this fashion.

TABLE I. INVERSE PROPORTIONALITY OF THE QUALITY OF THE SURF, MSER AND ASIFT DETECTORS VS. THE TIME TAKEN TO MATCH FEATURES

Detector	*Total Time (s)*	*Time per image (ms)*	*μ matched features*
SURF	36.5	9.7	15.0
MSER	45.45	12.0	9.1
ASIFT	4506.0	1192.0	65.6

B. Real-time Constraints

A further aspect of gesture recognition is the need for real-time image classification. In many cases these systems are being designed for use with sign language but there are also increasingly more examples of gestures being used for interacting with computerised surfaces. In either case the system must respond within a time limit. However, as one can see from Table 1 the response-times and feature matching success rates are verging on being inversely proportional.

Given an ability to on average compare 103 images per second using the SURF algorithm, a maximum frame rate of 10.3*fps* would be achievable if using a RANSAC [25] approach comparing only 10 images from a training set per classification. Similarly if ASIFT was used, it would not be possible to even achieve 1*fps* consistently without more optimisation which is beyond the scope of this work. If a system is to be expected to respond to human gestures in a

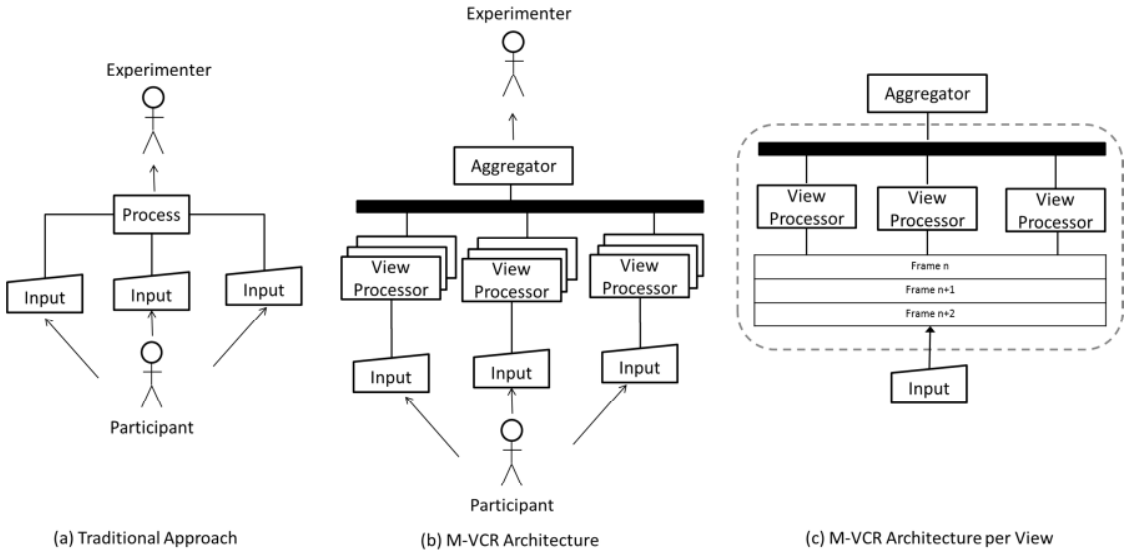


Fig. 2 M-VCR System Architecture. (a) demonstrates the traditional approach used in computer vision. (b) depicts the complete architecture used by the M-VCR approach demonstrating the three system layers. (c) shows an expanded view of the view-level processing

natural fashion, 1*fps* is not fast enough. In particular if the application domain involves sign language it would not be uncommon to require at least 4*fps* [4]. The MSER approach which detects regions rather points performs similarly to SURF in our benchmarking.

The M-VCR framework aims to allow the use of feature detectors that meet the necessary real-time constraints without as significant a degradation of the quality as is currently being seen.

IV. System Architecture

This section outlines the system architecture behind the gesture recognition process. The traditional architecture used in gesture recognition is first presented and critiqued following which the adopted architecture is presented and evaluated independent of the theoretical methodology of M-VCR.

A. The Traditional Architecture

Figure 2 (a) shows the simplicity of the traditional architecture which simply uses a single machine to process the incoming video streams and classify each of them either independently or in relation to each other. In this architecture input devices, which typically are cameras of various types, capture the hand gesture movements of a participant from various angles. Each input device is then either processed independently or combined into a single 3-dimensional object which is then classified. The classification results would then be provided to the experimenter.

The key point of failure is with reference to the benchmarking of the detectors that was depicted in Table 1. In the scenario shown in the figure there are three video streams which would result in a theoretical maximum achievable frame rate with the SURF feature detector of 3.4*fps* if the inputs are processed serially. By utilising the Matlab Parallelisation toolbox for the experiments, a frame rate of 10.3*fps* was consistently achieved when classifying several thousand images. Although this frame rate is satisfactory, the quality of the classifications using SURF and combining the data using weighted sums only successfully classified 42% of the images.

B. The Proposed Architecture

In Figure 2 (b) an overview of the proposed architecture for use in M-VCR is depicted. The architecture is composed of the following three system layers:

- Input Devices
- View-Level Processors
- Aggregator

As can be seen in part (c) of Figure 2 each individual input device publishes its data onto a 'bus' which is consumed by the allocated set of the view-level processors. The view-level processors each perform a classification as will be described in section V and publish this to the Aggregator. In the simplest scenario for each view only a single processor would be utilised, this would allow a frame rate of 10.3*fps*. By view we are referring to the algorithm process which consumes input data, not the individual input devices. As can therefore be seen in Figure 2(c) a single input device can feed into multiple view-processors via a publish-subscribe architecture. At the top level an aggregator reads in the data from all the view processors that supplied a result on-time. Those results supplied late are ignored.

As previously mentioned the architecture presented is inspired by the N-versioning fault-tolerance technique in [16]. At each view node the algorithms utilised for classification do not have to be the same, rather as emphasised previously utilising different classifications techniques can increase the reliability and confidence in the results. As previously mentioned, this approach is also building on the complementary fault tolerance technique of N-copy [23]which utilises data diversity rather than design diversity. A key element of the N-copy approach is the concurrency of data reads which are expected to occur at slightly different times causing slight differences in the data. The mapping from these varying inputs to a single output therefore remains the challenge. The proposed mathematical framework of M-VCR allows for not only 'n' inputs to be aggregated, but it also makes use of cumulative history state.

Experimental results will demonstrate how this N-version architecture increases the reliability of the classification for each view. Both approaches utilising independent instances of the same algorithm and different algorithms will be considered. Then in a similar fashion the results using multiple views will be presented.

V. Theoretical Methodology

This section documents the theoretical methodology and concepts behind M-VCR. It is therefore structured as follows:

- 'M' – The reasons for and approach to applying N-versioning in gesture recognition, building on the system architecture documented in the previous section.
- 'V' – Defining this level both in terms of representing a 'View' of the object but also a 'Version' of classification algorithm.
- 'C' – The consensus that must be reached in order for a final classification to be agreed upon.

This is then followed up documenting how this is applied in the gesture recognition scenario.

A. The need for 'M'

As has been eluded to the concept of N-versioning was derived from the need to decrease the susceptibility of a system to design faults in particular. As previously mentioned the choice of training set is a significant design decision. Therefore, although the concept of gesture recognition is in an entirely different domain to N-versioning; it is recognised that the classification of the same input image multiple times against 'different' training sets, or different subsets of the training set, equates to a different system design.

Traditionally in gesture recognition the training set is precompiled and new data is classified against the entire set at runtime. The use of RANSAC allows for only a subset of

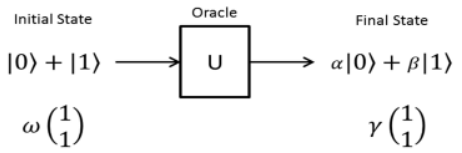


Fig. 3. Simple 2-state quantum machine showing both Bra-Ket and matrix notations

the training set to be used and given the random element each subset should be significantly different given a large enough set.

B. The duality of 'V'

In terms of M-VCR, 'V' can refer both to *'View'* and to '*Version'*. Initially it is considered as a *'View'* where the combination of data from multiple input devices can be combined to reach consensus, in a similar fashion to [5]. However, when considering *'Versions'* instead of views the concept mentioned in the previous section is embraced. Not only do we achieve different version designs by using RANSAC on the same algorithm but also by utilising different algorithms the design variation is increased. Further to the increased confidence in the results, this approach allows the speed of individual algorithms to be given greater priority in the trade-off against quality. For the remainder of this paper the term '*View*' will be used unless explicitly referring to the concept of multiple-versions for the same camera view.

C. Reaching Agreement

The method for reaching agreement is so specific to each and every situation that there appears to be no general consensus as to the best generic approach. In this subsection the novel mathematical framework of M-VCR is presented.

The concept of being in a superposition of states from quantum computing [26] has been adopted. This allows for a system to be in multiple states simultaneously and can be represented quite simply in matrix form. Figure 3 depicts a simple 2-state quantum machine applying some oracle or function to an initial state and outputting a final state in which the probabilities for each result are potentially different than at the input state.

In the M-VCR framework, at each view a ranked classification is achieved which is represented as a vector of probabilities whose sum is normalised. In our experimentation this is achieved by comparing the input image against a random sample from each class returning a measure of similarity. At the aggregator system level these vectors are then combined as a summation, at this stage to reach consensus there is no need for normalisation.

Further, in gesture recognition and in many other experimental situations it is normally not appropriate to simply recognise a single state but rather the sequence of states. There are therefore grammatical rules which dictate allowable sequences, regardless of what the gesturing is for. The approach of using normalised vectors representing the probability of each state allows for a matrix representation of the grammar itself being able to consist of probabilities of certain sequences. Consequently by matrix multiplication a classification can be achieved that takes into account:

- The sequence rules (R), an 'm' by 'm' matrix.
- The new view vectors of classification probabilities (VC_j) where there 'm' views.
- And the previous classification vector (C_n)

$$C_{n+1} = R \cdot C_n + \frac{1}{m}\sum_{j=1}^{m} VC_j \qquad (1)$$

The first half of the equation can be applied at either the view level or at the aggregator stage. In our experiments there was no conclusive difference between the approaches. It is expected that this may be a domain specific design decision.

This approach allows for any number of views to be used and the problem of not reaching consensus due to equally likely classifications is less likely due to each view providing a set of probabilities rather than a single result.

As can be seen in Figure 4 one of the key benefits of this approach is the utilisation of the history of classification probabilities for future classifications. Traditionally in systems that utilise classification history there is an audit trail of the decisions; however this is not built into the current form of the M-VCR framework. Rather, an accumulation of the history is represented as a single state vector. As such the effect is as if the audit trail was stored and utilised. In a theoretical case of having three classes: A, B, and C – this approach models the probability of being in any one of them

TABLE II. COMPARISON OF M-VCR AND WEIGHTED SUMS

	Weighted Sums	**M-VCR**
	$\boldsymbol{C_{n+1} = \sum_{j=1}^{m} w_j C_j}$	$\boldsymbol{C_{n+1} = R \cdot C_n + \frac{1}{m}\sum_{j=1}^{m} VC_j}$
Number of solutions	Each view provides a single solution, i.e. $\boldsymbol{C_j}$ is a single value	Each view provides a probabilistic ranking of solutions, i.e. $\boldsymbol{C_j}$ is a vector of probabilities
System Knowledge	Rules are fixed in the form of weightings applied to each view	A set of rules or a grammar can be expressed in a matrix independent of the views
Audit Trail	Separate approaches to creating and using an audit trail of classificaitons would have to be considered and methods for updating the view weightings would have to be applied	The next classification is affected by all previous classifications by the use of a cumulative history state

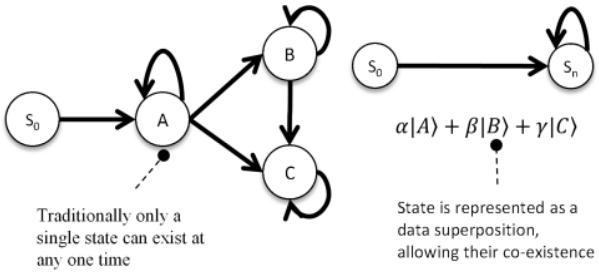


Fig. 4. Simple 2-state quantum machine showing both Bra-Ket and matrix notations

at the current point in time: $\alpha|A\rangle + \beta|B\rangle + \gamma|C\rangle$. α, β, and γ represent the probabilities of being in a particular state. In combination with a probabilistic rule-set, *R*, the conditional probability of a particular classification can be derived.

It is worth comparing this against current approaches, in particular against the method of using weighted sums. This comparison is outlined in Table II and it can be seen that M-VCR theoretically outperforms weighted sums. For the purposes of our experimentation all weightings have been left equal and the rule matrix has not been optimised for this particular scenario in terms of probabilities of certain classifications.

D. Our Scenario

Building on the theoretical reasoning stated previously Figure 5 outlines the software process that forms the basis of M-VCR. In part (a) of the figure the high-level process is outlined with more detail relating directly to the gesture recognition scenario for which M-VCR has been designed.

The experimental scenario is based on using gestures for simple interaction with a computer interface. At the very simplest level this would require the following gesture classes to be considered:

- Pointing at an object on the screen
- Clicking on the object
- Moving the object

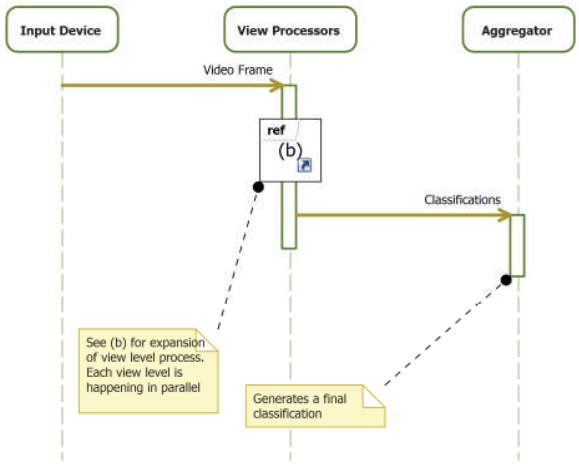
The third could be separated into the two sub-classes of dragging and then releasing an object at a particular location.

Considering therefore the sequence diagram of Figure 5(a) the View-Level processors consume a video stream of frames, and this frame rate is what is to be maximised in the trade-off against classification quality. This level of the system then performs the N-version processing as depicted in Figure 5(b) before publishing the resulting image classifications to the aggregator.

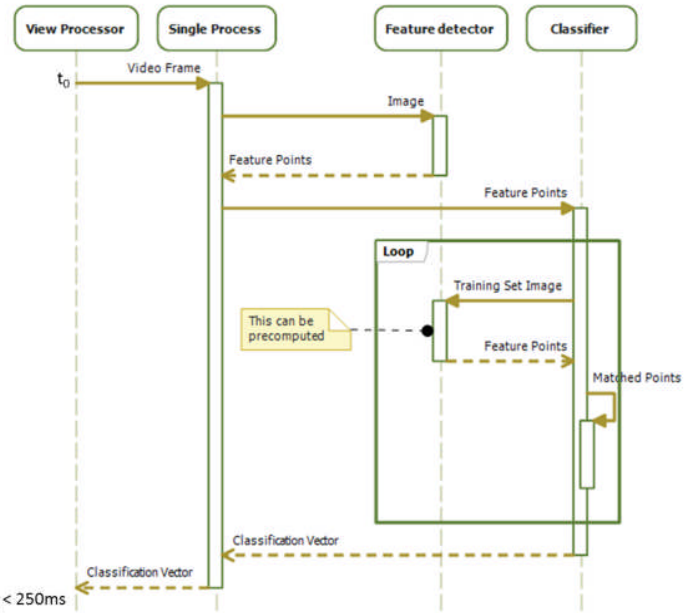
In the second sequence diagram the process at each version is portrayed. The feature detection algorithm is deliberately modularised away from the remainder of the process to allow for easier use of different approaches. This is similar for the classifier, although only a single approach is being considered in this paper. The key step for classification is the loop between the '*Classifier*' and the '*Feature Detector*' where the feature points of the input image are matched against those of the training set images. Acquiring the feature points from the training set data would traditionally be pre-computed meaning that this function becomes merely a look-up at runtime.

VI. Experimental Results

The hypothesis of this paper was to enable higher quality results in gesture recognition given strict real-time constraints, by aggregating results from low quality algorithms through the M-VCR framework. The experimental results in this section demonstrate how whilst consistently maintaining a video frame rate of 5*fps*, across all the views, the processor utilisation does not exceed 60%. Beyond meeting the real-time requirements, a classification



(a) M-VCR high-level process



(b) View level process which occurs in parallel on each view level processor

Fig 5. The M-VCR process. (a) summarises the process at a high-level. (b) picks out the stages of using a feature detector and then classifying the image against the training set.

success rate of 89% is achieved with algorithms that individually reach success rates of only 42%.

A. *Multi-View Consensus*

Of initial interest is the impact of utilising n-versioning for feature matching. Our gesture recognition experiments consist of up to four concurrent views being classified. The experiments run on video sequences at 5*fps* with frames at 640 by 480 pixels. The average video sequence consisted of around 200 frames to be classified. 100 video sequences were constructed with 20 different individuals participating to ensure significant variation. Specifically variation based on gender and ethnicity was enforced, particularly to capture skin colour and hand tilt variation.

Each view was independently processed, initially using just the SURF algorithm for feature detection, in order to guarantee meeting the real-time constraints. Across the four tested views, the average success rate of a single view was 42%. Each view had exclusive use of a single processor core with no lost frames at *5fps* with an average processor utilisation of below 49%.

By a simple aggregation of the various views according to the basic principles of n-versioning a classification success rate of 63% was achieved across entire video sequences, an improvement of over 20% on individual view results. Figure 6 shows the improvement trendline according to classification success rate alongside the increase in the number of views. Throughout the processor utilisation did not exceed 51%.

A question that arises in all instances where multiple concurrent versions of a system are required is that of cost. This is yet another aspect which in any real world system would act as a major attribute in a trade-off. Arguably there is a significant improvement by using even just two concurrent versions of 14%. However, the cost of having in excess of three versions would probably not be met by the increase in result reliability with results improving by a mere percentage point in subsequent system expansions.

B. *Multi-Version Consensus*

Our approach as previously stated goes much further than simply applying n-versioning to gesture recognition, rather as can be seen in Figure 6 by applying the consensus approach detailed in Equation 1 a classification success rate that would appear to be tending towards 90% was observed. This brought an average improvement on basic n-versioning of 21%.

Further, extending this to utilise a variety of algorithms, namely both SURF and MSER, in feature matching demonstrated a further 3% improvement, bringing results tending towards 93% successful classification over a video sequence. It is noted that the processor utilisation remained below 60% whilst running experiments at 5*fps* where each algorithm was processed on its own independent core.

It can be seen that when using only a single algorithm the results start tapering off around n=4, whereas with multiple algorithms the results taper off slightly later. In either case the cost of expanding the system beyond four concurrent versions would not be warranted in a gesture recognition system.

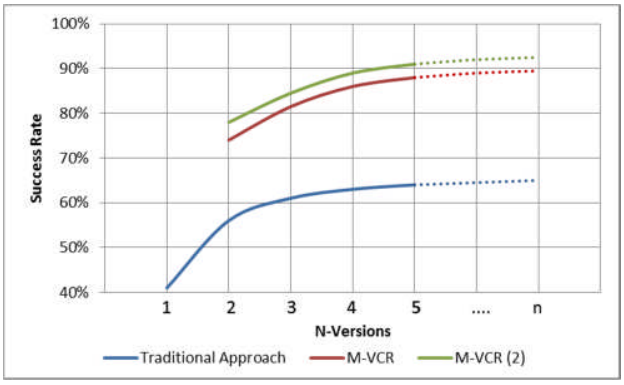


Fig. 6. Experimental results of M-VCR vs. the a traditional approach of Weighted Sums

We have experimentally shown a three stage improvement on traditional single view experimentation taking from an initial success rate of 42% to 89%:

1. 42% - Original success rate
2. 63% - After applying N-version and N-copy using Weighted Sums for result aggregation
3. 82% - After applying the M-VCR mathematical framework
4. 89% - After using different algorithms at the view level

That demonstrates an improvement of 47% on the traditional approach to image classification and 19% when using M-VCR over Weighted Sums. No degradation in the timeliness of the approach was observed. As a result by applying the M-VCR framework the achieved results with low quality algorithms are equivalent to those of much higher quality algorithms, whilst still meeting the real-time constraints of gesture recognition. This allows us to use algorithms of $O(n)$ rather than $O(n^2 + n^2 \log n)$ without a degradation of the results.

VII. CONCLUSIONS

This paper has proposed a framework for real-time aggregation of results from time intensive computations. The case study utilised has been in the realm of computer vision and specifically focussed on gesture recognition.

The M-VCR (Multi-View Consensus Recognition) approach proposed herein builds on the fields of n-versioning, n-copy as well as quantum computing and proposed in this paper allows for improved integration of experimental results compared to using weighted sums. Beyond experimentally and empirically evaluating n-versioning and n-copy to the realm of gesture recognition a novel result aggregation method is proposed that demonstrates a 19% improvement over using weighted sums.

This paper has experimentally demonstrated a significant improvement in successful results in the domain of gesture recognition. A total improvement of 47% was observed by utilising the M-VCR integration approach over running

independent and individual experiments as is the traditional approach in gesture recognition. Significantly, these results were achieved using O(n) algorithms that are able to meet the strict real-time requirements. The quality of the results attained within the time constraints were equivalent to those obtained by state-of-the-art $O(n^2 + n^2 \log n)$ algorithms.

Utilising this framework, the experimenter is able to include their knowledge of the system in rules which are expressed mathematically. Further although the approach does not explicitly keep an audit of the decisions, the cumulative history heavily influences the aggregation of the results by utilising the concept of being in a probabilistic super-position of all possible states at any one time.

VIII. Future Work

A further evaluation of the M-VCR framework for other scientific experimentations would be a valuable extension to this work.

Further analysis on the mathematical framework for aggregating experimental results will be conducted and more focus will be placed on the system architectures that would support such experimentation. Specifically the problem of an aggregation stalemate needs to be addressed such that decisions can always be reached. A stalemate would currently occur where the sum of the probabilities for different classifications are equivalent along with equivalent prior probabilities for those classifications.

The granularity of the timeliness investigation needs to be increased such that the framework can be correctly extended to provide optimisations at more specific points in the gesture recognition process. In our investigation the temporal considerations have been limited to entire view-level processors and the specific impact of competition for access to the input and output data queues was not considered. Therefore further investigation is required focussing on the temporal aspects of the data-flow. Of specific interest will be the scalability of the M-VCR framework with significantly more versions being aggregated.

Acknowledgment

The work presented in this paper was conducted in collaboration with Dr. Adrian Bors from the Department of Computer Science at The University of York, UK. Equipment and facilities for the experiments were provided by StormP Ltd. The work is supported by the National Basic Research Program of China (973) (No. 2011CB302602) and the UK EPSRC/JLR PSi programme.

References

[1] J. F. Abramatic, P. Letellier, and M. Nadler, “A narrow-band video communication system for the transmission of sign language over ordinary telephone lines.,” in *Image sequence processing and dynamic scene analysis*, 2nd ed., 1983.

[2] C. Yu, X. Wang, H. Huang, J. Shen, and K. Wu, “Vision-Based Hand Gesture Recognition Using Combinational Features,” in *2010 Sixth Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, 2010, pp. 543–546.

[3] N. Ahuja and M. Tabb, “Extraction of 2D motion trajectories and its application to hand gesture recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1061–1074, Aug. 2002.

[4] S. Fischer, D. Newkirk, and U. Bellugi, “Rate of Speaking and Signing,” in *The Structure of the Sign*, 1979, pp. 181–194.

[5] D. Kumarage, S. Fernando, P. Fernando, D. Madushanka, and R. Samarasinghe, “Real-time Sign Language Gesture Recognition Using Still-Image Comparison & Motion Recognition,” in *6th Int. Conf. on Industrial and Information Systems*, 2011.

[6] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, “Pfinder : Real-Time Tracking of the Human Body,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, 1997.

[7] N. Shimada, K. Kimura, and Y. Shirai, “Real-time 3D hand posture estimation based on 2D appearance retrieval using monocular camera,” in *Proceedings IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, 2001, pp. 23–30.

[8] Y. Iwai, H. Shimizu, and M. Yachida, “Real-Time Context-based Gesture Recognition Using HMM and Automaton,” in *IEEE Proc. of the Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, 1999.

[9] G. He, S. Kang, W. Song, and S. Jung, “Real-time Gesture Recognition using 3D Depth Camera,” in *IEEE 2nd Int. Conf. on Software Engineering and Service Science*, 2011, pp. 187–190.

[10] T. Starner, J. Weaver, and a. Pentland, “Real-time American sign language recognition using desk and wearable computer based video,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1371–1375, 1998.

[11] A. Avizienis, J. Laprie, B. Randell, and C. Landwehr, “Basic Concepts and Taxonomy of Dependable and Secure Computing,” *IEEE Trans. DEPENDABLE Secur. Comput.*, vol. 1, no. 1, 2004.

[12] J.-C. Laprie, “Dependable computing and fault-tolerance,” *Dig. Pap. FTCS-15*, vol. 2, no. 11, 1985.

[13] A. Avizienis, J.-C. Laprie, and B. Randell, “Fundamental Concepts of Dependability,” 2001.

[14] B. Littlewood and L. Strigini, “Validation of ultrahigh dependability for software-based systems,” *Commun. ACM*, vol. 36, no. 11, pp. 69–80, Nov. 1993.

[15] L. Chen and A. Avizienis, “N-version programming : a fault-tolerance approach to reliability,” *FTCS-8*, vol. 3, pp. 3–9, 1978.

[16] A. Avizienis, “The N-Version Approach to Fault-Tolerant Software,” *IEEE Trans. Softw. Eng.*, vol. SE-11, no. 12, pp. 1491–1501, Dec. 1985.

[17] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” in *British Machine Vision Conference*, 2002, pp. 384–394.

[18] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, Nov. 2004.

[19] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF),” *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[20] G. Yu, J.-M. Morel, and J.-M. M. Guoshen Yu, “Asift: a new framework for fully affine invariant image comparison,” *SIAM J. Imaging Sci. 2*, vol. 2, pp. 438–469, 2011.

[21] J. P. Lewis, “Fast Normalized Cross-Correlation Template Matching by Cross-,” *Vison Interface*, vol. 10, no. 1, 1995.

[22] J. J. Horning, H. C. Lauer, P. M. Melliar-Smith, and B. Randell, “A Program Structure For Error Detection And Recovery,” in *Operating Systems*, 1974, pp. 177–193.

[23] P. E. Ammann and J. C. Knight, “Data diversity: an approach to software fault tolerance,” *IEEE Trans. Comput.*, vol. 37, no. 4, pp. 418–425, Apr. 1988.

[24] T. Anderson and P. A. Lee, *Dependable Computing and Fault-Tolerant Systems: Principles and Practice*, 2nd ed. Springer-Verlag, 1990, pp. 1–313.

[25] M. A. Fischler and R. C. Bolles, “Random Sample Consensus : A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] P. A. maurice Dirac, “The principles of quantum mechanics,” *Int. Ser. Monogr. Physics, Oxford Clarendon Press*, vol. 1, 1947.