Fish4Knowledge Deliverable D7.6

Final report to EC - Technical

Principal Author: UEDIN
Contributors: All partners
Dissemination: PU

1 Executive Summary

The study of marine ecosystems is vital for understanding environmental effects, such as climate change and the effects of pollution, but is extremely difficult because of the inaccessibility of data. Undersea video data is usable but is tedious to analyse (for both raw video analysis and abstraction over massive sets of observations), and is mainly done by hand or with hand-crafted computational tools. Fish4Knowledge developed methods that allow a major increase in the ability to analyse this data: 1) Video analysis that automatically extracts information about the observed marine animals which is recorded in an observation database. 2) User interfaces that allow researchers to formulate and answer higher level questions over that database without needing specialist programming skills.

The project concept was to acquire undersea video data from up to 10 cameras off the coast of Taiwan in coral reef areas, detect and track fish observed in the videos, which are then recognised according to their species. A database recording the data extracted from all processed videos was created. A user interface was developed that allows marine ecologists to assess the distribution of fish by time, date, species, and location. All project goals were achieved.

As an indication of achievement, the project recorded 524K unique videos, each 10 minutes long, resulting in 87K hours of video (91 Tb). From these, 1.4 billion individual fish instances were detected, which were tracked, resulting in 145 million trajectories. 45% of the videos have had their fish recognised (starting from the highest quality videos first), resulting in 57 million recognised fish so far. The SQL tables to record the processed results required about 400 Gb.

The project published 46 peer-reviewed publications so far, plus 6 MSc dissertations, and is expected to lead eventually to 5 PhD theses. Project code is stored in SourceForge. Fish detection, tracking, recognition and unusual behaviour ground truth data are publically available. A subset of the raw videos and the full processed results are publically available. The user interface is publically available at:

`http://gleoncentral.nchc.org.tw/`.

We promoted our approach to data capture and analysis to the marine ecology community. A partner project has started with the CaribSave consortium involving some of the technology and expertise developed here.

2 Project Context, Objectives and Achievements

The project was designed as a next generation big data experiment, in which the data feed was live video observing undersea coral reef formations and fish (as contrasted with most previous video analysis that observes people). The justification for this project concept was that it would push the research boundaries in the ability to 1) remotely record and store video data, 2) detect, track and recognise objects in a difficult visual environments (water and illumination distrubances, uncontrolled targets, unbalanced species composition), 3) present large amounts of extracted noisy information is a manner usable to marine ecologists, but without requiring them to be computer programmers, and 4) process and store the data acquired in a flexible and efficient manner. The project was aimed at 'big data', whereby the project would acquire an image database: c. 2 billion frames, from which we extracted 1+ billion fish (images recoverable but not explicitly stored) and their corresponding descriptors (explicitly stored), on the order of the world's largest image databases (Google had 10+ billion in 2010, Flickr has est. 7 billion in 2013).

The original project objectives as stated in the proposal were:

1. Detecting targets in noisy environments.
2. Characterising interactions between the targets.
3. Recognising fish species by integrating multiple 2D perspectively distorted views over time.
4. Exploiting ontologies to interpret user queries.
5. Exploiting ontologies to convert queries into workflow sequences.
6. Storing and accessing massive amounts of video and RDF data in a timely manner.
7. Integration of the research in a publically usable web tool.
8. Creation of a fish database suitable for behavioural and environmental studies.
9. Training of staff in cross-disciplinary methods (computer vision with database and workflow scientists, computer scientists with biologists).

All of these objectives were achieved, except that the data ended up being stored as SQL instead of RDF (although an RDF interface was developed to allow access in RDF form).

The more detailed objectives of project year 3, which were all achieved, were to:

1. Enhance the detection and tracking algorithms.
2. Extend the species recognition algorithm to more species and higher accuracy.
3. Complete system integration (workflow and user interface)
4. Evaluate system performance

5. Enhance system to increase data analysis and query answering speed
6. Evaluate usability by marine biologists
7. Process project year's 1-3 previously recorded videos (Detection complete, Recognition in final processing stages, 45% complete at time of writing).

The main public outputs of the project were:

1. Algorithms and associated software for: target detection in complicated environments, video quality classification, fish species recognition.
2. A database of 1+ billion detected, tracked and recognised fish covering 23 species, which represent 99+% of the observed fish (about 500 Gb).
3. A subset of the raw videos and associated extracted results (about 1 Tb).
4. 46 papers, 5 MSc dissertations and eventually 5 PhD dissertations, all open access.

The key achievements/discoveries/innovations of the project were:

1. **Image Analysis**: Background modeling results beyond the state of the art, both in underwater videos and in standard datasets (e.g., I2R).
2. **Image Analysis**: Novel approach for discriminating objects of interest from the background, which extends significantly the objectness approach by including motion features.
3. **Image Analysis**: A covariance particle filter able to handle multi-object occlusions and to track effectively objects with 3D complex and unpredictable trajectories.
4. **Image Analysis**: Novel methods for efficiently acquiring large scale ground truth using clustering.
5. **Image Analysis**: Novel methods for recognising deforming similar shapes (fish) in 3D under variable lighting conditions, taking advantage of temporal consistency, and overcoming a large imbalance in the class sizes.
6. **User Interface**: Novel approaches to presenting the ground-truth evaluation, and their impact on user trust.
7. **User Interface**: Initiated a study of novel methods for identifying and presenting potential biases in data.
8. **Workflow**: Novel methods for tracking and controlling computation progress in a complex, but fallable, multi-processor/multi-resourcecomputing platform.
9. **System**: A novel interface between the datastores and the heterogeneous compute machines was developed. The project also devised a novel framework to integrate processes and data within this infrastructure.

10. **System**: A massive amount of ecological video was recorded. Without explaining details of duplicated video content, the raw video storage (91Tb) covered:

Resolution	<5 fps	5-8 fps	9-23 fps	24 fps	>24 fps
320x240	5,520	189,101	5,383	0	0
640x480	0	90,653	12,356	264,421	1,117

Based on the research achievements, the most valuable future research directions are thought to be:

1. **Image Analysis**: Exploiting foreground knowledge of the tracked objects for better detection, and exploring the benefits of higher resolution and faster sampling.
2. **Image Analysis**: exploring the benefits of higher resolution on recognition.
3. **Image Analysis**: Developing methods for more efficiently acquiring ground-truth.
4. **Image Analysis**: Developing methods for estimating the correctness of results when only a tiny proportion of the data can be manually evaluated.
5. **Workflow**: Investigation into a more sophisticated self-monitoring and self-repairing workflow.
6. **User Interface**: Developing methods for conveying the correctness of results from massive data sets.
7. **Integration**: Methods for obtaining GroundTruth for 'big data' problems and how to validate the GroundTruth as representative.
8. **Integration**: Creating workflow methods for monitoring progress in massively parallel and failure-capable process execution.
9. **Integration**: Methods for improved communication of results between independent processes and teams.
10. **Computational System**: Investigation into methods for communication and resource control across heterogeneous architectures.

Below are expanded summaries of the achievements of the individual workpackages.

2.1 WP 1: Video Data Analysis

Fish detection and tracking are two key components of the F4K system as they aim at turning the raw video data into information processable by the downstream components such as species recognition, user interface and workflow composition.

The underwater domain has several difficulties that make the tasks of fish identification and tracking very challenging and all the strategies adopted within the F4K project have been influenced by the following factors:

Table 1: Categorisation of the quality of the video dataset

Classification	Number (1000s)	Percent
Normal	75.8	14%
Complex Background	37.4	7%
Algae on Lens	49.4	9%
Blurred Water	182.0	35%
Highly Blurred Water	65.0	12%
Encoding Errors	108.1	21%
Unknown	6.2	1%
Total	524.1	100%

- **Sudden light changes** mainly due to the light propagation in water as affected by the water surface shape;
- **Multimodal backgrounds and periodic movements** (e.g. plants affected by flood-tide and drift) which may lead to misclassify background areas as target objects;
- **Low-quality videos** in terms of image resolution and video frame rate, due to bandwidth limitations between the cameras and the storage servers;
- **Image quality**: atmospheric phenomena (e.g. typhoons, storms), murky water and bio-fouling generally affect the quality of video frames, thus making the video analysis components more prone to errors. Image compression errors also affected many videos.;
- **Appearance model**: as fish have three degrees of freedom and undergo erratic movements, their shape is subject to sudden changes (further amplified by the low video frame rate);
- **Motion model**: Besides the difficulty introduced by the low video frame rate (which caused fish to move by a significant amount of pixels between two consecutive pixels), fish' motion patterns are typically hard to understand and predict.

Based on this observation, we developed an algorithm that categorised the type of the video. Based on a ground truth sample, we estimate the accuracy of this process at about 93%. After processing all 524K videos, we categorised the videos as shown in Table 1. Notably, only 75K (14%) of the videos are 'normal'.

However, the main constraint that we had to take into account, beyond the ones mentioned earlier, was the **computation time**: as the fish detection and tracking modules had to deal with continuously-recording videos and with a huge amount of previously-recorded clips (dating back to 2009), they could not afford to spend too much time on processing a single video (also because they were the upstream modules and could represent a bottleneck for the entire system). For this reason, the fish detection and tracking algorithms were designed in order to balance the trade-off between accuracy and efficiency.

2.1.1 Fish Detection

Detecting fish in videos is the first fundamental task of the F4K system. This task has been carried out by resorting to background modeling approaches – as opposed to template matching (not applicable because of the large variability of fish appearance) and motion analysis (low-resolution videos do not allow an effective estimation of fish motion model) methods – which aim at building an estimated image of the scene without objects of interest; this model is then compared to each new video frame for identifying foreground objects.
First, we tested several pixelwise state of the art approaches, which were previously tested under conditions recalling the ones present in our underwater scenes [7]. These approaches, basically, model the pixel's history through an explicit background model that may consist of either a mixture of probability density functions or a temporal median. More details can be found in Deliverable 1.1. Although on the initial ground truth dataset, these approaches performed fairly well, when more complex scenes were taken into account, their performance dropped dramatically leading us to investigate other solutions. In detail, their main downsides were in the adopted background model and in background update mechanism which were not suitable to deal with the peculiarities of the underwater domain. As a consequence, two other solutions were adopted: 1) the first one was inspired by the original codebook approach [5], which, however, showed many limitations with videos 320×240 at 5 *fps*, because it requires, in the training phase, a long sequence of "stable" background images; 2) the second one was inspired by the VIBE approach [2] and models the background pixels with a set of neighbourhood samples instead of with an explicit pixel model (see Deliverable 7.5 for a thorough description). Spatial influence of neighboring pixels is also taken into account and the background update mechanism is simply based on a uniform pdf. Performance evaluation reported in D5.4 showed that this approach represented a good compromise between efficiency and accuracy and, as such, it was used for processing the whole set of historical videos.
Year 3 was, instead, devoted to investigated methods for improving accuracy. In particular, by following the current research trends in background modeling, we developed a new detection component which relies on a domain-range kernel estimation approach and that models not only the background pixels but also the foreground ones. The method, moreover, uses information on neighboring pixels and employs textures (namely, the Texton [6]) robust to illumination in the modeling process. The performance evaluation (see Deliverable 5.5) showed a significant improvement in accuracy not only in the underwater domain but also in other scenarios outperforming the most recent approaches. Of course, the increase in accuracy was achieved at the expenses of efficiency as the new method is about one hundred time slower than the one used for the production runs. A qualitative comparison of this last approach, and the *VIBE-like* method is presented in Fig. 1, which shows that our approach had high qualitative performance.

Generally, we did not use any post-processing to improve fish detection results but removed the connected components whose area was lower than a certain threshold set empirically (as a percentage of the input frame) as the value below which it was not possible for a human to distinguish the colour and texture of the objects. Spatio-temporal regularisation was also

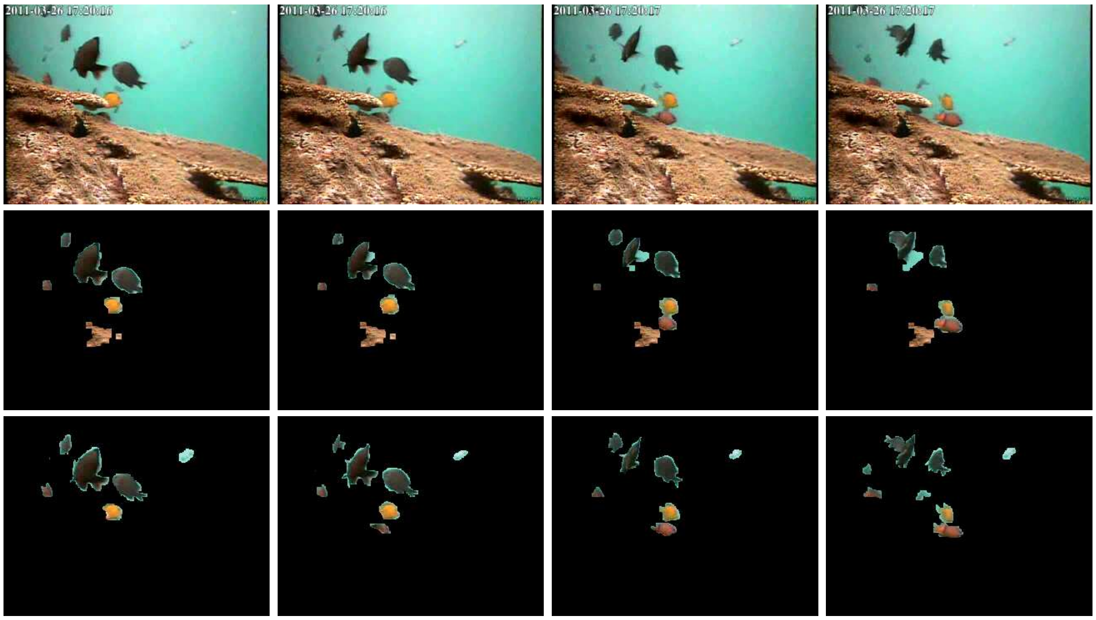

Figure 1: Qualitative Comparison of background subtraction methods (from top to bottom): 1) (first-row) video-frames, 2) $VIBE-like$ (second row), which detects parts of rocks as fish because of light changes and 3) our kernel-density estimation approach (last row) which is able not only to reduce false positives (rocks not detected) but also to detect tiny fish (bright spot on the right hand side) whose appearance looks like the background's one.

investigated to improve the quality of the segmented objects; though it allowed us to achieve better accuracy, its application lowered sensibly the algorithm's efficiency (also memory issues arose) making the whole system two orders of magnitude slower.
In order to reduce the fish detection computation time, we implemented a GPU version of some of the above algorithms. In particular, the GPU version of the VIBE-like approach (the one used for production run) was able to obtain a speed-up of about 70 times (other algorithms could be accelerated at most by 10-15 times) on a single 2496-core machine. Good as they were, these results implied the need for at least 10 such machines to equal the processing power provided by NCHC's 1000-core cluster. Moreover, fish detection is not the most time-expensive part of the system, surpassed by far by the fish tracking and fish classification parts, which, algorithmically, provide less opportunities for parallelisation.

Finally, we also introduced a module estimating the uncertainty in the fish detection process. In fact, moving elements in the scene (apart from fish), such as plants and algae, and the sudden luminosity changes due to sunlight gleaming on the water surface or on the seabed and rocks, may be misclassified as fish, thus providing misleading information to the higher levels. In order to deal with such cases, this module assigns a quality score to each detected object by estimating the likelihood that the detected object is a fish using a-priori knowledge on shape, colour, boundary and motion. In detail, features such as contour complexity, colour contrast and optical flow differences from the object's surroundings, correspondence of scene segmentation and edge detection with the object's contour, and internal homogeneity (in terms of colour and optical flow) are computed and then passed to Naive Bayes classifier, which provides the probability that the detected object is a fish. We also evaluated the use of homogeneity criteria [3] (based on the internal colour variance of the fish region), background keypoint matching and kinematic features extracted from a global affine motion model [4], but our analysis showed that these did not provide a significant contribution to the uncertainty computation. Although, in principle, these latter features might help to model better fish appearance and motion, again, the low quality of the available videos made them application useless.

As future work, we will focus on two aspects: 1) exploiting the generated big visual data (about 10^9 detections) to improve the detection (as well as the recognition) performance by using simple nearest-neighbor approaches and 2) reducing the processing times by devising suitable hashing techniques for speeding up the feature extraction, the model update process and, eventually, the post-processing phase.

2.1.2 Fish Tracking

We present the two tracking approaches devised to extract fish trajectories: a *covariance tracker* and a *covariance particle filter*. Both approaches share the way fish appearance is modelled, which is as the covariance matrix of a set of multimodal pixel-based features (location coordinates, RGB and hue colour values, directional derivatives) [8]. This model allows for an elegant way to merge spatial and statistical features – as well as their correlations – into a compact format. A covariance metric was also adopted to compare such representations in a way more suitable to their mathematical nature than simpler and more intuitive approaches (such as the L_2 norm).
The initial *covariance tracker* employed this model to compute a description of objects provided by the detection module, in order to match them in consecutive frames, based on their covariance similarity. The main limitation of this approach was the absence of a motion model whatsoever, so the search area was established in a heuristic way by averaging the distances of previous objects in that video. Moreover, the approach was strongly dependent on the accuracy of detection module; therefore, whenever the latter failed to detect a fish or when occlusions happened (which caused two or more fish to be merged into a single blob), the tracker would fail as well. In order to overcome these limitations, we devised and implemented a particle filter framework which integrated both the covariance modeling approach and the information coming from the detection module. We use a first-order motion model (since we do not aim at modeling the motion in a complex way, based on the peculiarities of the processed videos) to update the particles' position, and the covariance metric is used for the particle weight update process. Also, more weight was assigned to particles which overlay motion areas (as detected by the previous module). In order to keep the computation time low, we only use ten particles per object, which proved to be enough to follow an object accurately. An advantage of the particle filter is that we do not need to evaluate empirically an object's search area, since this is done implicitly by the motion model and the presence of several particles for each object, which allow to cover and analyze a larger area. Nevertheless, the low video quality (in terms of

spatial and temporal resolution) affected the tracker's performance. In fact, the relatively low video frame rate (5 to 11 frames per second) causes objects to move substantially in consecutive frames, thus requiring the search area (for the original covariance tracker) or the particles (for the particle filter) to spread out in a wider area. This implies that other similar fish may be included in the search area, and, given both the low spatial resolution of videos and the fact that in underwater images, colours fade as objects move away from camera, misclassification might occur.

Finally, a separate discussion concerns how we handle occlusions. The covariance tracker basically performs tracking-by-detection, so it associates objects by means of a similarity metrics based on the covariance distance. If multiple objects happen to fall within a fish' search area, the one with the closest covariance representation will be selected as "new location in this frame" for the target object. However, this only works as long as fish do not overlap in the frame: in this case, the fish detection module fails to identify them as separate instances. On the other hand, the covariance particle filter only uses the output of the fish detection as hints on location hypotheses, so in principle it can tell fish apart even if they are partially overlapping. However, if the area over which a fish is located is too small, the covariance model becomes less sensitive (because the covariance matrix is computed from a smaller set of feature vectors) and there is the risk that the tracker associates the overlapped object to part of the overlapping object. For this reason, when the tracker detects occlusions (as partial overlaps between two fish' bounding boxes, after each has been independently tracked), it acts depending on the degree of overlap: if it is high (more than 25% of the smallest bounding box's area), we temporarily "disable" one of the objects (the one with the highest covariance distance) until the occlusion is resolved; otherwise, we track them normally. Fig. 2 shows two cases of fish-fish occlusion effectively handled by the covariance particle filter.

Figure 2: Fish-fish occlusions handled by the covariance particle filter. Let us note that the tracker is able to distinguish between fish with similar appearance and that move consistently in consecutive frames.

The covariance particle filter allowed us to overcome some limitations of the covariance tracker (see Deliverable 5.5); however, the improvement in performance was achieved only when high resolution videos (the ones gathered within the AQUACAM research programme) were considered, while with the F4K videos, the two trackers performed almost the same. Our future work on tracking will be developed along two lines: 1) to infer from the huge amount of trajectories extracted by processing the historical video dataset a reliable fish motion model for the particle filter prediction step; and 2) to develop a multi-camera tracker able to merge and mutually verify the tracking information coming from cameras with overlapping field of views.

In conclusion, the *VIBE-like* and the *covariance tracker* (as they represented the best trade-off between accuracy and efficiency) were used for detecting and tracking fish in the whole historical video dataset, which amounted, as of July 2013, to 535,345 ten-minute videos (from 5 to 11 *fps*). The processing took 70 days using 600 cores 24 hours a day and resulted in about 1.5×10^9 fish detections (for about 10^8 different fish) and the whole database size consisted of about 300 GBytes (from 92 TBytes of initial raw video data). However, a wide set of detection and tracking algorithms together with their performance on different video classes and image regions was also made available for being used by the workflow, so that the best combination of algorithms can be selected in case of on-demand video processing. All the fish detection and tracking approaches devised to investigate improved accuracy were greatly influenced by the low quality of the available videos (almost 70% of videos had problems, see Deliverable 5.5 for details): in fact, the low frame rate made impossible to estimate a reliable fish motion model, while the low spatial resolution had an impact on the fish appearance computation. Despite all these difficulties, the achieved results are satisfactory. Please note that all the components were tested on annotated datasets built with the crowdsourcing and collaborative tools described in Deliverable 5.6.

2.1.3 Fish Recognition

The project developed the Balanced-Guaranteed Optimized Tree with a Reject option (BGOTR) to filter less confident recognition results. Since our data are obtained from a challenging underwater environment where the fish are freely swimming, the recognition results contain classification errors that can be mainly categorized into three types: false detections, misclassified samples of the BGOTR method, new species of fish that are not included in our ground-truth dataset. We apply a Gaussian Mixture Model (GMM) as the reject option to reduce the error rate.

Fish recognition was implemented as a 23-species BGOTR hierarchical tree. This tree is automatically constructed from a heuristic method based on the inter-class similarities. It applies feature selection at each node for better presenting the samples into an optimized feature space where a multi-class SVM classifier is trained. The performance estimates are based on 27370 fish images from the top 23 species (Figure 4) with a 5-fold cross validation. The training and testing sets are isolated so fish images from the same trajectory sequence are not used during both training and testing. We developed a trajectory voting method as an improvement to minimize the environmental influence.

We compare the performance of BGOTR (Average Recall, 75.26%) against the flat SVM classifier (69.81%). After feature selection, the SVM method has been improved (70.62%). PCA

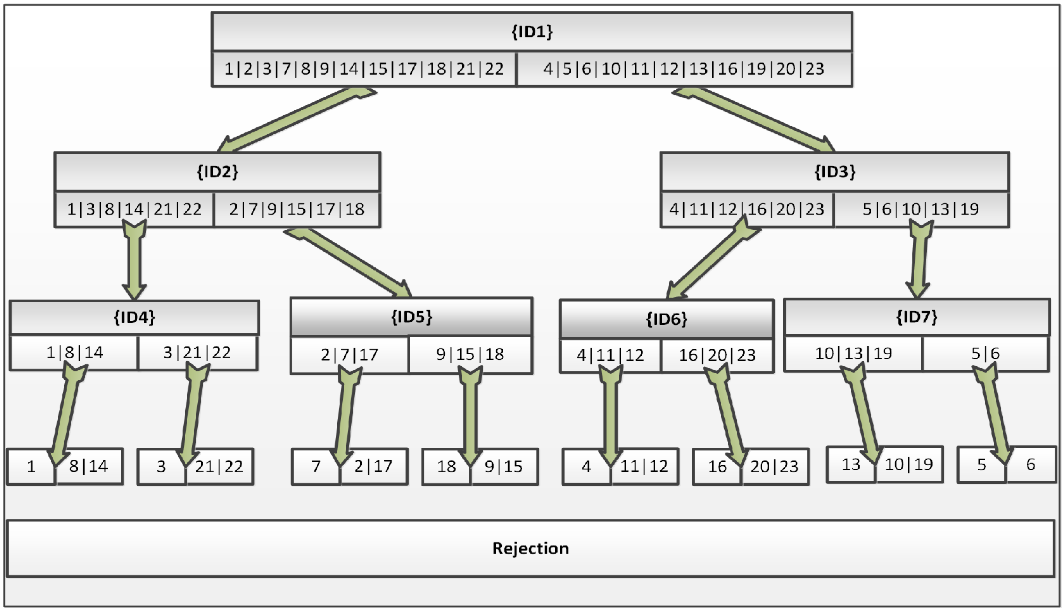

Figure 3: Result rejection in fish recognition.

is a popular alternative approach to reduce feature dimensions and achieves better performance (73.53%). We evaluated the reject option with 24150 fish images of the top 15 most common species. Since our database is imbalanced and only the top species have adequate samples to train the rejection model, we only apply the reject option to the top 6 species. Additional minority species (8 species, 3220 fish images) are included in the test set to test the performance in probing new species. Our method rejects a significant portion of the misclassified samples (True Rejection, TR) while the cost is that it also rejects a smaller proportion of correctly classified samples (False Rejection, FR). We compare our GMM-based method with two state-of-the-art methods (Table 2) and achieve significant improvement.

Algorithm	F_1-score
BGOT+SVM probabilities [9]	0.7150 ± 0.0222
BGOT+soft-decision hierarchy [10]	0.7140 ± 0.0225
BGOTR	$\mathbf{0.7485} \pm \mathbf{0.0194}$ *

Table 2: F-score of the top 6 species after rejection. * means significant improvement with 95% confidence.

2.2 WP 2: Interactive User Query Interface

2.2.1 User Interface Main Achievements

The main achievements of the User Interface workpackage are:

Figure 4: Ground-truth data: 23 species.

1. Enhanced current user community data collection practices.
2. Established user information needs suited to being addressed by automated video analysis. These include user scenarios for establishing and sharing fish species abundances to motivate design rationale for developing system user interfaces.
3. Developed a design rationale for user interfaces supporting both types of needs.
4. Developed a method for evaluating video analysis ground-truth data tailored to marine biology/ecology experts.
5. Developed an intermediate user interface for comparing video analysis errors for marine biology/ecology experts, requiring interpretation of the measures used in the video analysis community.
6. Refined the set of user information requirements for both species abundance exploration and uncertainty visualization based on intermediate system and user interface implementations.
7. Made user interface refinements for exploring fish species abundance and conveying and controlling uncertainty measures. Our user interface design and implementation contributes to the HCI domain and a wider range of use case beyond F4K, including: (1) novel visualizations of video analysis performance tailored for non-experts' needs: communicating classifiers' performance to end-users could be facilitated by our simplified performance visualization; and (2) novel interaction techniques for data exploration: data exploration efficiency could be improved by our multi-purpose interactive visualization.
8. Developed a working end-to-end web environment that provides high-fidelity access to all data provided at the back-end, integrates workflow functionality and functioning implementations of user interface designs.

2.2.2 Establish user information needs

The user needs of the community have been studied since the writing of the proposal and throughout the course of the project. We used the current status of the data and the user interface at each iteration of interaction with potential users not to try to perfect a single user interface, but to understand how the emerging system could be constructed to meet their current and future scientific needs.

Initial user requirements on the types of information they would like to publish on are reported in deliverable D2.1 *User Information Needs* [15]. These include measures based on fish abundance, with information related to species.

During the project it became clear that the measures supplied by the system on fish counts were highly unreliable, in particular in terms of the quality of the video data collected and the "true" numbers of fish derived from the measures in the system. This led to a shift in emphasis on the user interface work from a comparatively straight-forward design effort on how to create a user interface for counting the detected fish, to a more complex investigation of how potential users understand the uncertainties inherent in the system and, despite these, how they would still be able to draw scientifically valid conclusions.

Biologists from Taiwan and the Netherlands, from a wide range of research fields, have been involved in our user interviews and experiments, namely: coral reef fish, pelagic and demersal fish, corals, plankton, microorganisms and ecotoxicology.

The ground truth collection has resulted in insights into the extent to which professional marine biologists are able to identify fish species consistently, which is not always possible because of both video quality and visual distinctions per species. This has been translated into game-like user interfaces that encourage lay users to participate in identifying fish species, with the attempt to reach at least the same agreement as the experts, to then be used for increasing the ground truth set available for the video components in the project.

2.2.3 Explore component-based prototypes

In D2.3 *Component-based prototypes and evaluation criteria* [13] we identify the types of uncertainty information that need to be communicated to the end user to allow them to understand the relationship between what the system is able to provide and the information needed by the user. We discuss the quality of the ground truth data obtained with the user interfaces built for this new purpose. We also gives examples of both basic and more advanced user interfaces that are able to communicate (aspects of) provenance and implicit and explicit uncertainty information, either visually or via an interaction dialogue.

D2.3 *Component-based prototypes and evaluation criteria* [13] presents a series of mockups that guide the implementation of the user interface in the third year of the project. These mockups give consistent interfaces for tasks marine biologists want to carry out, specifically to allow selections of location and period and to obtain analyses of the counts of fish. The

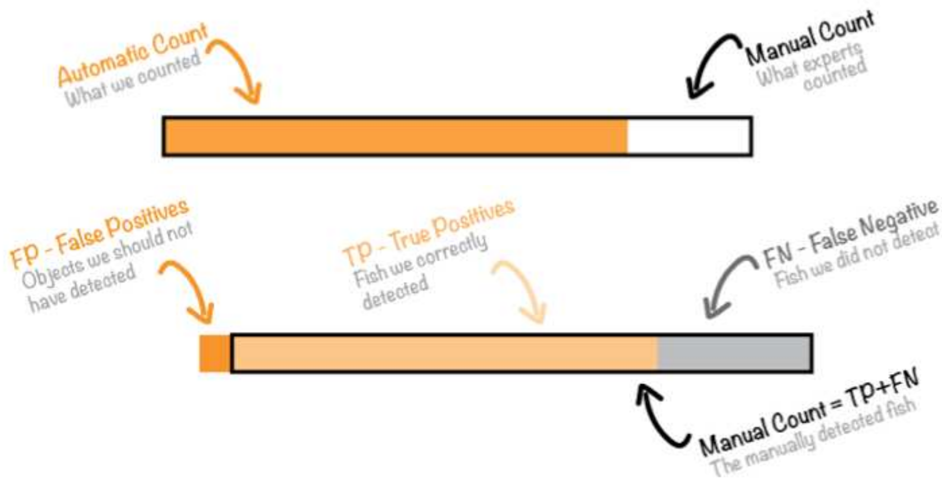

Figure 5: Experimental designs for visualizing ground-truth based evaluations. These designs offer simplified versions of the advanced visualizations used by computer vision experts. We evaluated how such visualizations support user trust, fulfill their information needs, and remain understandable ([11]).

system has been built to support these queries and at first sight is relatively straightforward. The complexity of both the underlying system design and its visualization is in estimating the counts based on the results of the video analysis components, and on the ground truth evaluation, and in conveying these in a way that the marine biologists will trust the results (see example in Fig. 5). We experimented with the visualization of ground-truth based evaluations, prior to evaluating an end-to-end system populated with the video analysis results. Our findings, published in [11], highlight the difficulties for conveying technical concepts of image processing to non-experts, and for addressing user needs for extensive provenance information which span beyond the report of ground-truth based evaluations.

2.2.4 Evaluation and in situational user testing

Video analysis tools have been introduced relatively recently to this community and no well-accepted data analysis framework has been set up for the usage of video data for marine biology research. Our user studies, summarized in this section, provided valuable insights for understanding the potential usage of our tool, and, more generally, for understanding the acceptance of video analysis tools by the marine biology community ([11, 12, 14, 15]).

The types of evaluation that are well-accepted by the image processing community are not easy to understand by marine biologists (e.g., ROC evaluation). In our studies the majority of biologists encountered difficulties with understanding the technical concepts. Thus it is difficult for them to evaluate the potential errors introduced by computer vision components. We observed that users tend to overlook the technical details that can bias their analysis. They

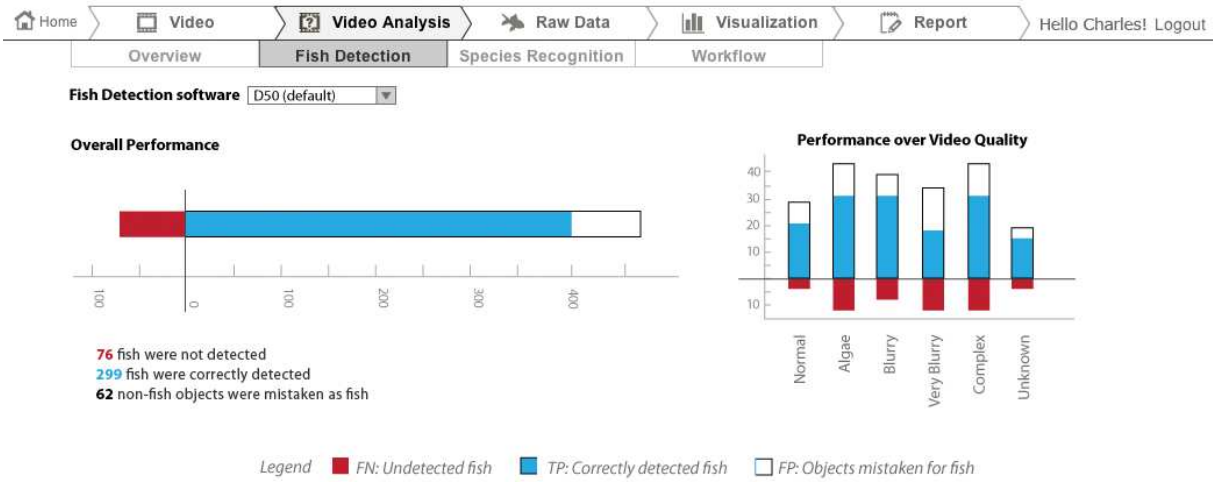

Figure 6: **The Video Analysis Tab - Fish Detection sub-tab** provides visualizations of ground-truth based evaluation of the Fish Detection components. Evaluations are provided for each video quality (e..g, Blurred or Normal videos). The *Species Recognition* sub-tab provides evaluations of the Species Recognition components.

also do not perceive the software as fully reliable, and expect large numbers of errors, as well as biases (e.g., systematically larger error for specific species or video quality). However, we found that biologists are still likely to accept the tool for their research for two reasons. First, video analysis tools can considerably reduce the effort currently involved for manual annotation of videos. Second, biologists are used to dealing with the high level of uncertainty in the collected data (e.g., fishery data, diving observations), since underwater ecosystems are difficult to access, and are often impossible to observe directly (e.g., open sea, deep sea). The most important user feedback concerns the following issues:

Provide understandable validation of the video analysis software: The technical methods used to validate the tool could be difficult to understand and accept by the marine biology community. Therefore, they suggested using methods adopted from biology (e.g., counting fish in a controlled environment, repeating measurements). They also were eager to trust the image processing expert opinion while choosing the settings for the software (e.g. the most reliable version of the software to detect particular species). Addressing this feedback led to the visualizations shown in Fig. 6. They provide the exact numbers of True Positives, False Positives and False Negatives as classified in the set of ground-truth items. They do not display True Negatives or rates (e.g., True Positive Rates or Precision). The design is intended to reduce user cognitive load by i) reducing the amount of information displayed, ii) avoiding confusion or irrelevancy due to the fact that True Negatives are introduced by the Fish Detection components, but are not present in the ground-truth dataset, ii) avoiding misunderstanding of advanced mathematical representations (i.e., of rates such as True Positive Rates or Precision).

Provide comprehensive provenance information: Regarding uncertainty issues, biologists expressed requirements for technical information other than ROC-like evaluation:

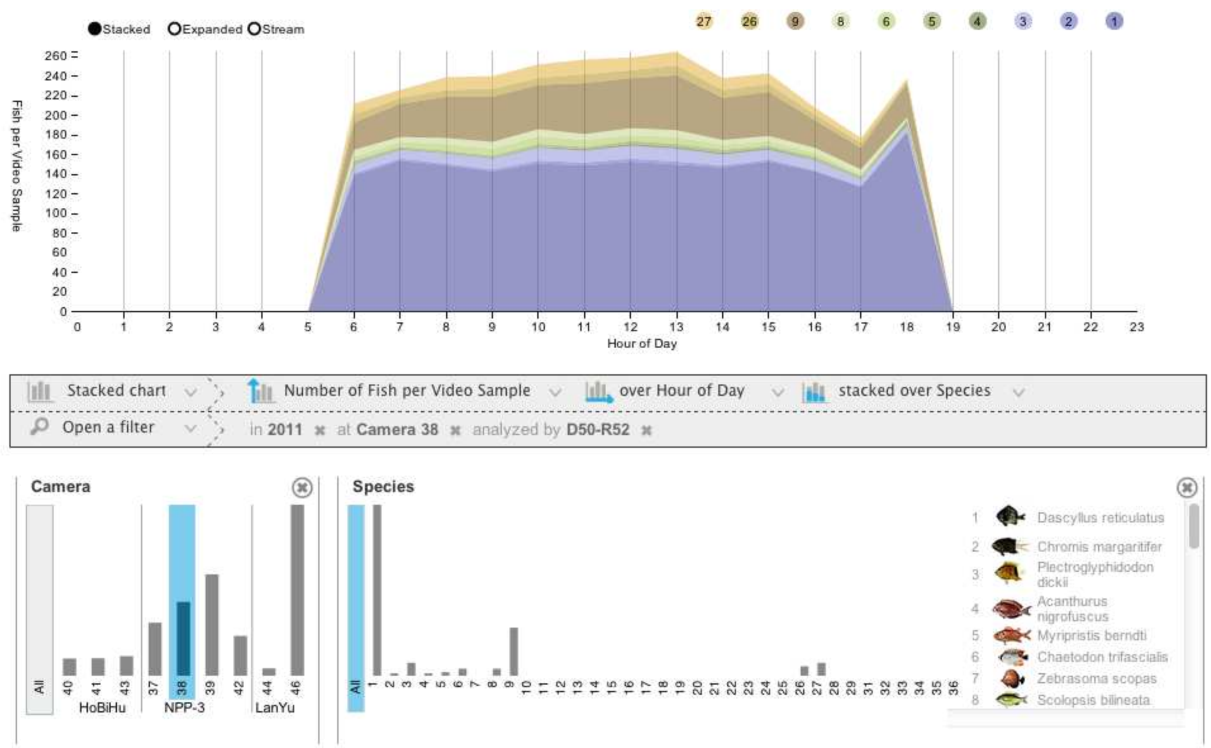

Figure 7: A visualization of the species composition display. The upper display shows the fish per video sample for the period 06:00-18:00 (actually for all 24 hours, but the cameras only record in the daylight hours). The bottom portion of the display shows what options have been selected for the display, in this case all species are selected (blue bar at right) and only for camera 38 from site NPP-3 (blue bar at the left).

- The image quality of the video samples used (e.g., fuzziness, murkiness). Video quality may bias the video analysis results. For instance, seasonal events like typhoons can influence video quality, and thus the seasonal abundance patterns observed. We addressed this issue by providing a filter widget allowing the user to select datasets with a specific video quality (Fig. 10).
- The performance of the video analysis components for various video qualities (e.g., more errors may occur with murky videos). We addressed this requirement by providing the visualizations shown in Fig. 6.
- The rate of duplicates of single fish in fish counts. Some species may produce more duplicates than other species, because of their natural swimming patterns (e.g., residential fish swimming back and forth the cameras' field of view). This is a potential bias for studying the relative abundance of each species (e.g., species composition). We were not able to address this complex issue, which requires the collection of more diver-based observation data.
- Description of the habitats observed within a camera's field of view (e.g., the species of coral). We satisfied this requirement in the current system by allowing users to view the video in which the fish have been recognised (the *Video* tab) and, hence, also the habitat at the location. We do not yet provide a description of the surrounding habitat just outside the camera view.

Locations of the cameras: The coverage of the ecosystem of study is essential and specific to every research topic. Many biologists want to choose the location for their cameras individually. Additionally, they would like to have a service that could process videos captured by cameras independent from F4K. Such videos could be recorded in transects, e.g., with a moving background. Several biologists are interested in taking this further internationally.

High-level information needs: A number of additional visualizations and UI features were suggested by users, such as: the integration of solar and lunar calendars for filtering datasets of interest, or the usage of the traditional data analyses used for biodiversity research. Further investigations are needed to support the choice of relevant, general-purpose biodiversity metrics[1].

2.2.5 Support of user needs

The user information needs have been collected during the project in D2.1 [15], D2.2 [12] and D2.3 [13]. The user scenarios developed in D2.2 [12] illustrate the expected use of the system based on these needs. These include the information needs for exploring the collected data, controlling the execution of video analysis components on a specified set of videos, and explaining the likely number of fish counted by the system (i.e., its uncertainty) based on ground truth evaluation of the video analysis components.

[1] Some example can be found in http://en.wikipedia.org/wiki/Diversity_index

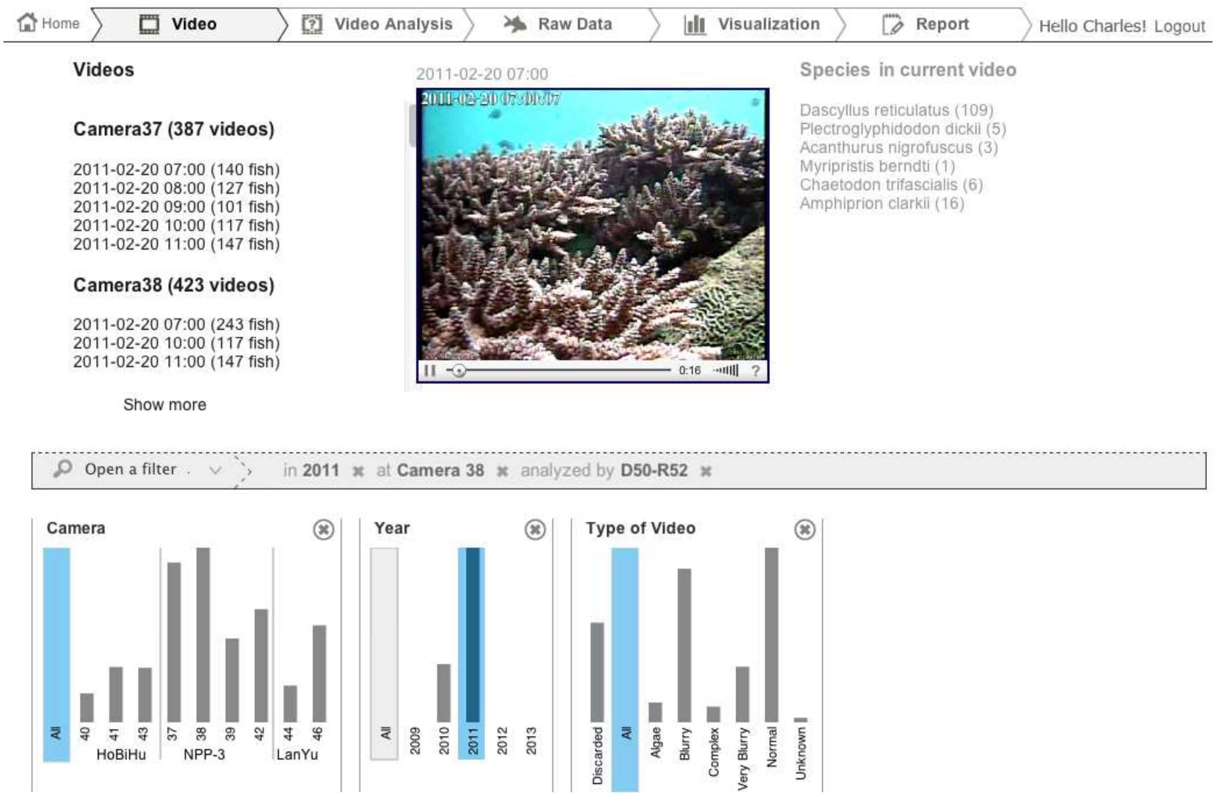

Figure 8: The Video tab. In this figure, the top field shows the selected video, so the biologists can observe the actual videos as well as the processed results. The species detected in the video clip are listed at the right. The bottom blue fields report the selected video source, year and catagory of video. The selected videos are listed at the top left.

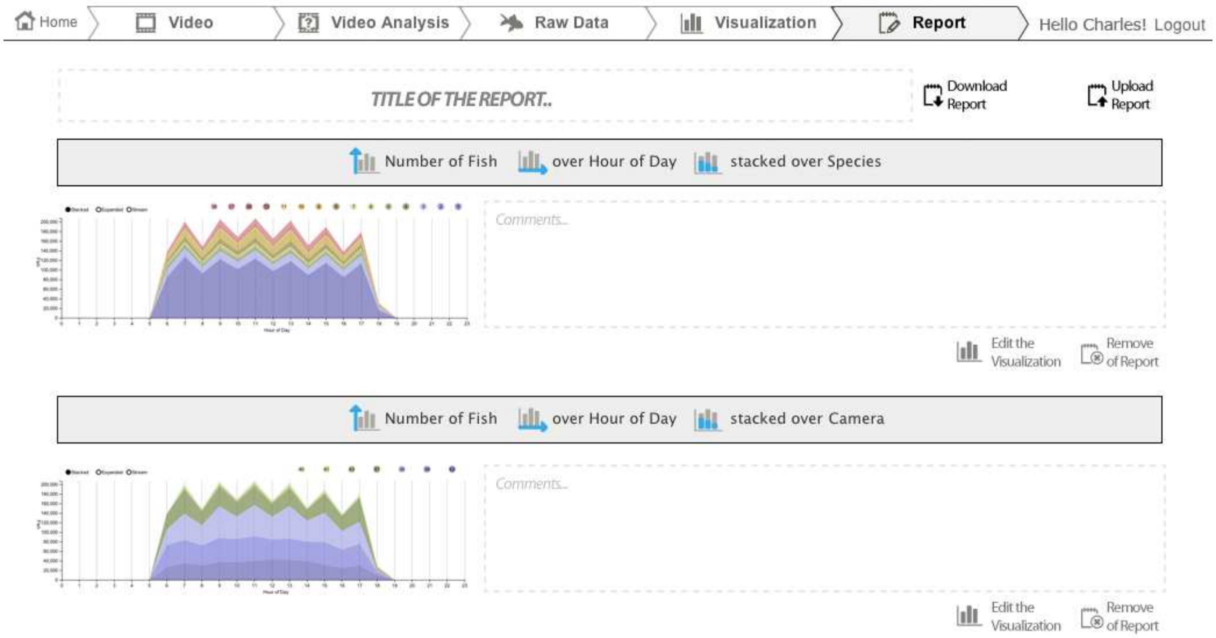

Figure 9: The Report tab. The statistics is similar to Figure 7 except the plot shows aggregation over species (top) versus camera (bottom).

The user requirements are synthesized in D2.3 - Appendix I [16] and addressed in the final version of the public query interface, namely:

1. **D2.1-A - Support the analysis of population dynamics**: We support this requirement by providing the following metrics: abundance (e.g., fish counts), species composition (e.g., fish counts stacked species), species richness (e.g., number of species). These metrics can be calculated on user-specified datasets targeting specific fish populations. The functionalities for visualizing species composition are shown in Fig 7.

2. **D2.1-B - Support the browsing of videos of interest**: This requirement is addressed by the features of the *Video tab* (Fig. 8). Users can browse the videos that correspond to the data that are currently visualized.

3. **D2.1-C - Support the identification of trends and correlations of trends**: We support this requirement by providing users with comparisons of fish count by using stacked charts (Fig. 7), and/or by gathering visualizations for specific areas using the *Report* tab (Fig. 9).

4. **D2.3-A - Expose the uncertainty of video analysis components**: This requirement is addressed by the *Video Analysis tab* (Fig 6). It provides users with ground-truth based evaluations for each video analysis component.

5. **D2.3-B - Estimate the errors contained in the visualized datasets**: Multiple factors potentially impact the errors in the video analysis results. This requirement is supported

Figure 10: A Visualization of the numbers of video samples available for March 2011. The lower part of the interface shows the distribution of videos per Camera, Year, and Type of Video Quality.

by 3 functionalities: i) each fish of the visualized dataset is given a *certainty score*, which can be used to filter out fish with high chances of being False Positives; ii) the visualization of the *Numbers of Video Samples* let users evaluate if the sampling size is sufficient (e.g., fewer videos lead to more uncertainty); iii) the visualization of video quality (Fig 10) let users evaluate if image quality impact the observed results, according to the ground-truth based evaluations for each video quality (Fig 6).

2.2.6 End-to-end system integration with data

In D2.5 *UI components integrated into end-to-end system*, we describe the implementation of the User Interface component and its connection with the database storing the video analysis results. The architecture uses the *Model-View-Controller* paradigm, and state-of-the art web-based visualization libraries (*D3*, d3js.org). The connection with the workflow ensures a balance between the automatic and continuous processing of the videos, and the dynamic assignment of top-priority video processing addressing specific user needs.

2.2.7 Future Work

The improvement of the public query interface will continue after the end of the F4K project. The usability of our proposed uncertainty visualizations and multi-dimensional visualizations will be evaluated and refined. Further research of interest concerns the study of uncertainties other than errors of video analysis software (e.g, varying camera's field of view, duplicated individuals), methods other than ground-truth based evaluations (e.g., risk of confusing species), and methods for fish count normalization (e.g., usage of the logistic regression technique, based on our fish certainty score).

However, the delivered User Interface for exploring video analysis data, and its potential uncertainty, constitute a first attempt for introducing such data collection technique within the marine biology domain. As it is, the tool is useful for both marine biologists, who for the first time can explore continuous data streams, day after day, and year after year, and for computer scientists, who can demonstrate the opportunities of video analysis techniques and investigate potential refinements of their usage.

2.3 WP 3: Process composition and execution

The workflow component of the F4K project is responsible for investigating relevant methodologies and implementing a working workflow system towards the end of the project. More specifically, its task is to take in video data that has been captured by the F4K project partner NARL and analyse and process them in useful ways to answer targeted user queries. The approach that we have chosen for the workflow system is illustrated in a three-layered knowledge-based framework: the upper Design Layer, the middle Workflow Layer and the lower Processing Layer.

Based on descriptions of domain data, user goals and partner system components, the workflow manager selects suitable software modules based on dynamic user queries and run them in a complex High Performance Computing (HPC) environment. As this HPC environment is a shared facility that we do not have full control of and also the fact that some elements of it are experimental, over time, we experienced unpredictability on job execution quality, performance and continuity. In addition, the quality of the video data (that are captured from open sea) as well as their absence (e.g. due to disruptions caused by bad weather) also contributed to some of the poorer performance. Poor performers may result in jobs hanging in queues without being detected or handled, leaving their depending jobs starving. Other examples are jobs seemingly being executed successfully, but no results have been generated nor stored. Such jobs, without being properly tracked and handled, may be lost indefinitely and their results missing.

During the 2nd and 3rd year, to ensure the smooth execution of the system and results properly recorded, we have worked very closely with the Video and Image Processing (VIP) teams to understand their algorithms and in particular to raise flags for abnormalities in the database, as appropriate, so that we can track VIP modules executions. In addition, we have created an error detector and handler to monitor the execution status of each jobs in the queues, as they are spawned from user queries. We have also worked very closely with the HCI team, so that the user is kept informed via the HCI interface.

To facilitate the above error handling, we have extended our domain ontologies, inc. performance metrics, to enable our workflow system to better deal with performance issues in a systematic manner. To work with performance issues, we have created several additional database tables to store job monitoring and error handling information. Detailed usage of these tables is in the 2nd year report. During the 3rd year, we have provided additional definitions and usage of such tables for the UI team, inc. the query_management table, error definitions and error handling algorithms. Based on these newly devised mechanisms, we can derive the performance of software components more reliably and thus utilise them better.

In conclusion, the workflow team has achieved its targets and beyond. This document focuses on reporting efforts made during the last year, drawing on our previous efforts. This includes extensions of our earlier efforts in domain ontologies that is a part of the 3-layered knowledge framework; the workflow system development and integration efforts of the F4K system; F4K system performance analysis, error detection and handling mechanisms. To understand the performance of the integrated F4K system, we have further collaborated with Prof. Omer Rana (University of Cardiff, UK) and Dr. Rafael Tolosana (Universidad de Zaragoza, Spain) to provide a more detailed analysis using the *Quality of Resilience* framework.

2.3.1 T3.1 - Extensions to Domain Ontologies

Task 3.1 created a set of suitable domain ontologies that are based on user requirements for our intelligent workflow system. This work is also to be coordinated with the system user interface specification work that is described in WP2. We completed such a set of initial ontologies by the end of year one. This set of ontologies is further improved over time, as we discovered additional user and system requirements. Terminologies defined in these ontologies have been used by the workflow and partner systems for communication purposes, primarily through the database.

In particularly, the original F4K domain ontologies were enhanced with concepts related to system performance measurements and improvements made in the final year of the project. We included both hardware- and software-related measures that would inherently help improve the overall performance of the workflow and F4K system when considered appropriately. Figure 11 depicts this extension.

2.3.2 The Computing Resources Sub-Tree

The F4K computing environment is a heterogeneous platform made up of a group of virtual machines (VM cluster) and a supercomputer (Windrider). The VM cluster contains nine nodes with machines of different specifications. Windrider consists of seven queues also of different

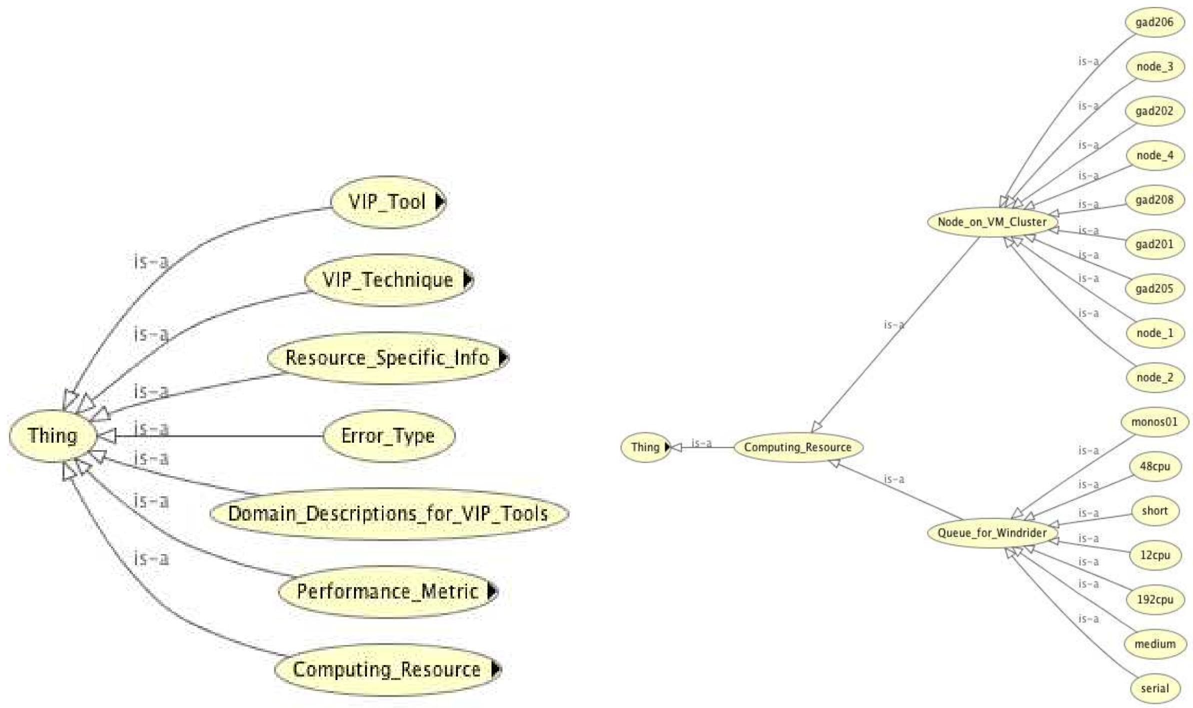

Figure 11: (Left)Higher level concepts of the Capability Ontology now include ‘Computing Resource’, ‘Performance_Metric’, ‘Resource_Specific_Info’ and ‘Error_Type’. (Right)The ‘Computing Resource’ class and its subclasses in the Capability Ontology.

capacities. On the VM cluster, jobs are distributed onto nodes based on their availabilities by a dispatcher. Available nodes are, e.g., node 1-4 and gadX. On Windrider, jobs are managed through several shared queues. Based on a prior performance analysis of these queues according to our usage requirements, we have distributed our jobs onto several most suitable queues: monos01, serial, short, medium and long. They are represented in Figure 11.

2.3.3 The Performance Metrics Sub-Tree

The performance of a software component that is queued, executed and monitored on a resource can be measured using several performance metrics. This typically includes the average waiting time on a queue, average execution time on a machine, the maximum and minimum execution time, overall success rate (to execution completion) and average database waiting time. We have therefore included these performance metrics in the performance metrics branch of the ontology as shown in Figure 12. Table 3 gives an example output based on the performance metrics. Component 135 and 136 outperformed its peers. They are therefore the default choices, if the user has not provided a preference.

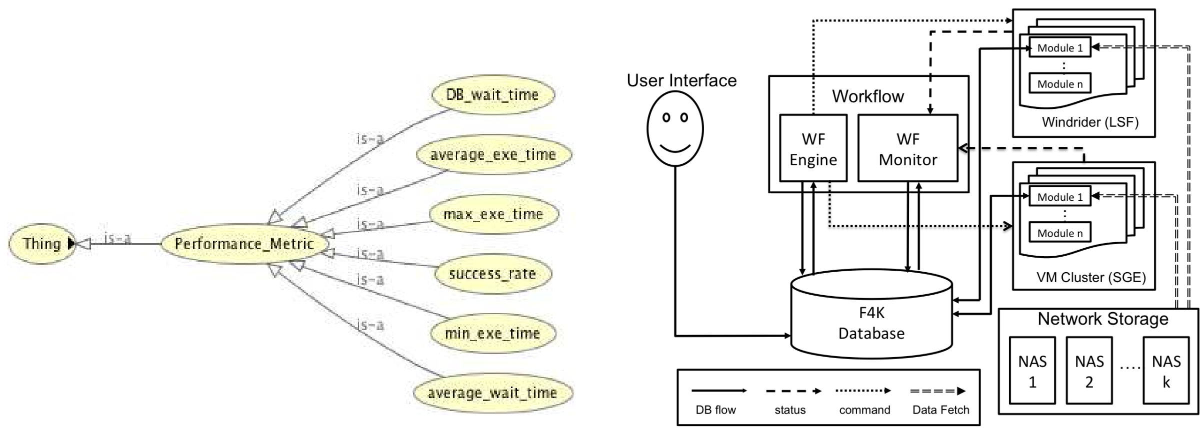

Figure 12: (left) The addition of the 'Performance_Metric' class and its subclasses to the Capability Ontology. (right) The workflow system communicates with other components via database or command line. It consists of a workflow engine that generates abstract & concrete workflows, and a monitor that handles errors and provides statistics.

Table 3: The performance metrics of the stable software components for fish detection and tracking (IDs 135, 141 and 142) and fish species recognition (IDs 128 and 135) in F4K. * denotes default component.

Component ID	Avg. Execution Time(s)	Avg. Queuing Time(s)	Max. Execution Time(s)	Min. Execution Time(s)	Avg. DB Wait Time(s)
128 recognition	8796 (∼2.4hrs)	6164 (∼1.7hrs)	355381 (∼4days)	15	68
135* detection	734 (∼12mins)	90 (∼1.5mins)	19604 (∼5.4hrs)	0	93
136* recognition	9902 (∼2.75hrs)	42655 (∼11.5hrs)	344113 (∼4hrs)	16	32
141 detection	892 (∼14.7mins)	31460 (∼8.7hrs)	2845 (∼47mins)	10	4
142 detection	11336 (∼3.15hrs)	53205 (∼4.8hrs)	28107 (∼7.8hrs)	180	11

2.3.4 T3.2 - Workflow System Design

Task 3.2 created a design of a workflow system of two layers based on user specifications and domain ontologies defined in T3.1. Such an initial design was reported in the first year's report that includes the above two layers and an additional layer - the design layer. This design was further extended and improved over time. During the 2nd and 3rd year, it was extended to accommodate new project developments, inc. the use of database tables as a communication means, the revised control and data flows, and changes in the HPC environment. It also adds a new error detecting and handling sub-system that is a part of the workflow system. Figure 12 (right) depicts this new architecture.

To understand the performance of the workflow system, we list errors that have occurred within the F4K workflow execution and compare their impacts on the system depending on whether the error handling mechanism is in use or absence in Table 4. It is obvious when the system does not make use of the workflow's error handling component, suitable resources and queues are not being selected and used. Jobs that fail are not being re-run and in extreme cases some jobs may starve or be suspended indefinitely. These factors affect the overall F4K system and HPC computing performance and thus other user's jobs.

Table 4: Performance of Error Handling System

Scenario	System Handling using Workflow	System Handling without Workflow	Possible Effect(s) without using Workflow
Successful Job	Finished	Finished	All jobs are waiting in the same queue without utilising full system capability
Failed Job	Re-run once	Exit directly	The failed job will not be detected until a manual check is performed
Job dependency	With dependency handling	Without error handling	The dependent job could be queueing forever
High priority job waiting too long	Suspend low priority job to release resources	Job waits in the queue	High priority jobs can be held for a long time
Low priority job waiting too long	Resubmit with higher priority	Job waits in the queue	Low priority jobs can be starving in the queue

2.3.5 T3.3 - Intelligent Workflow System and QoR

Task 3.3, based on system design in T3.2, developed an intelligent workflow system. We have implemented such a system that is an integrated and central part of the F4K system. In addition, we evaluate the system performance based on a 'Quality of Resilience' framework by calculating the likelihood of failure when using different combinations of VIP modules to achieve the same task.

"Quality of Resilience" (QoR) [1] is a metric that identifies *how* resilient a given workflow *is likely to be* prior to its enactment. Consider a workflow in F4K to detect, track (subquery Q1) and identify (subquery Q2) all fish species over a given date range and set of camera locations. The planner will generate a workflow template consisting of two data dependent steps: ($t1$) for detecting & tracking and ($t2$) to identify fish species. Currently, there are 4 candidate abstract tasks for $t1$ and 2 for $t2$. The planner uses detection accuracy and performance as a criteria for selection between them. So far, F4K has registered over 600,000 executions of this query (using one instance of $t1$ and $t2$). This enables us to use this data to understand the QoR associated with this workflow. Table 5 summarises QoR values for all possible instances of $t1$ and $t2$. With

Quantitative QoR Metrics Classification: QoR_U: task level							
metric	description	$t1_1$	$t1_2$	$t1_3$	$t1_4$	$t2_1$	$t2_2$
m1	number of alt. tasks	3	3	3	3	1	1
m2	number of input tasks	0	0	0	0	1	1
m3	number of resources	1	1	1	1	1	1
m4	task failure rate	3.02	4.12	6.0	2.1	21.7	12.3
m5	task exec. time (secs)	397	411	1596	1342	4984	13134

Table 5: QoR Metrics: Task-level

QoR Metrics: workflow-level ($wf_1 = t1_1 + t2_1;\ wf_2 = t1_1 + t2_2$, etc.)									
metric	description	wf_1	wf_2	wf_3	wf_4	wf_5	wf_6	wf_7	wf_8
m6	avg. number of alt./tasks	2	2	2	2	2	2	2	2
m8	number of task joins	1	1	1	1	1	1	1	1
m9	wf failure rate	12.36%	7.66%	12.9%	8.21%	13.85%	9.15%	11.9%	7.2%
m10	wf exec. time (secs)	2550	5244	5371	4747	7857	9726	5531	14643
m11	overall number of resources	2	2	2	2	2	2	2	2

Table 6: F4K Compilation of Quality of Resilience Metrics at workflow level

4 different tasks for $t1$ and 2 for $t2$, up to 4*2=8 different workflow variants are generated. For each workflow variant, a QoR metric is provided in Table 6.

The Quality of Resilience (QoR) metrics give a detailed account of system performance at both the task and workflow level. While $t1_4$ and $t2_2$ being the best performing modules in their categories (lowest failure rate). The best combined failure rate is $wf_2 = t1_1 + t2_2$ 7.66%. This option however is not the fastest. For the future, it is therefore very interesting for the workflow planner to learn to adapt itself at the run-time according to user requirements and changing system circumstances to reach an optimal solution.

2.4 WP 4: High Performance Storage and Execution Architecture

The goal of WP4 was to develop a sustainable infrastructure to support scientific discovery in the field of marine ecology. The infrastructure is composed of three interconnected components: up to 10 video cameras continuously recording and sending data streams, a massive storage system to store raw and analysed data, and a high performance computing facility to do data analysis. By year 3, we integrated these components into a seamless platform which supports the development of knowledge discovering processes within this project. The main activities in year 3 for WP4 were: (1) to improve the stability of video capturing and transmission, (2) to extend storage and computing capacity, and (3) to improve the database read/write performance.

2.4.1 Video capturing and transmission

As part of the project, we redesigned the architecture of the observation system to support capturing of high resolution videos. In an earlier version, videos were transmitted to the storage site directly. There was a risk of data lost caused by network instability. In the new design, we added a set of hardware components as a first line processing and buffering device which stores videos temporarily while the video transmission process is waiting for network channel. Requested by project partners, we also tested on encoding video with higher bitrates (5M). These higher bitrate videos provide clear and reliable data source for further video analysis. With the buffering space we are able to tune the video source to send higher resolution videos, which was impossible in earlier model of direct network transmission due to limited bandwidth. Figure 13 shows the new architecture of observation system.

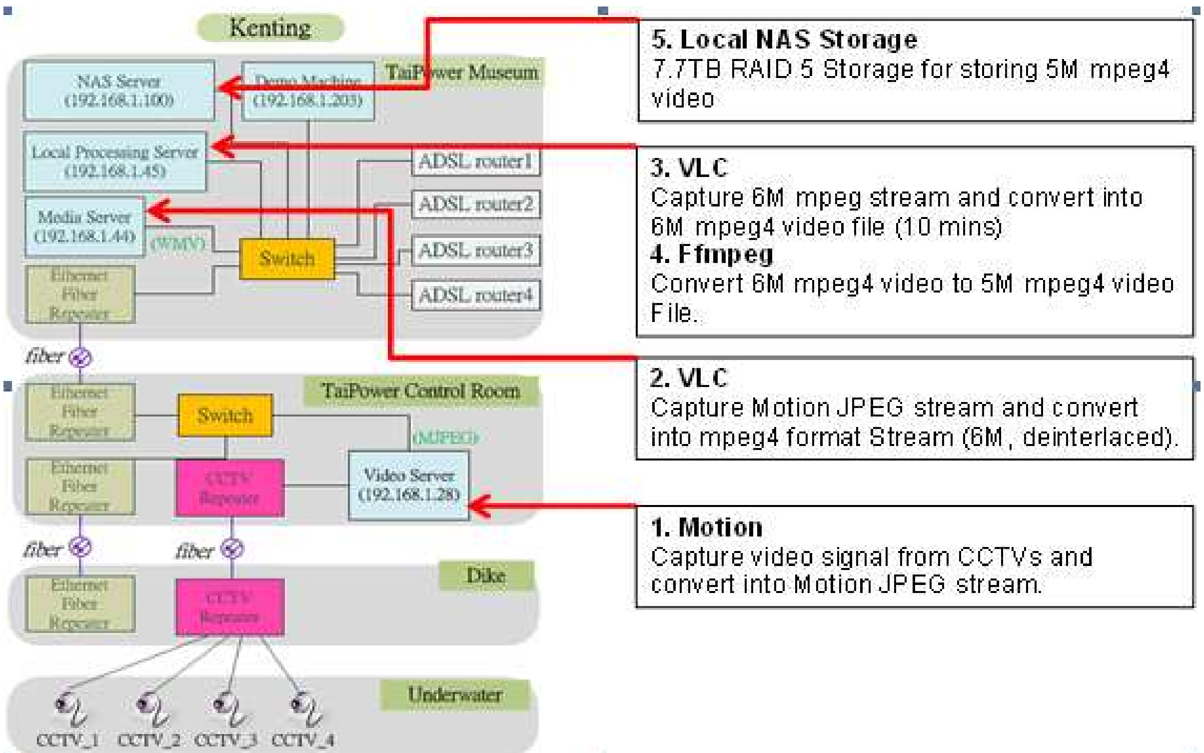

Figure 13: New architecture of observation system

The new added component is also capable of doing first line video processing like conversion of encoding format. It is used to convert raw video stream into other encoding formats, like H.264, MPEG4, etc., at high bitrates. In this case we choose to encode the video in the MPEG4 format based on computing time consideration, because it takes six times more computing time to encode in H.264 than in MPEG4. Figure 14 shows the process flow of video data from capture device to final storage facility.

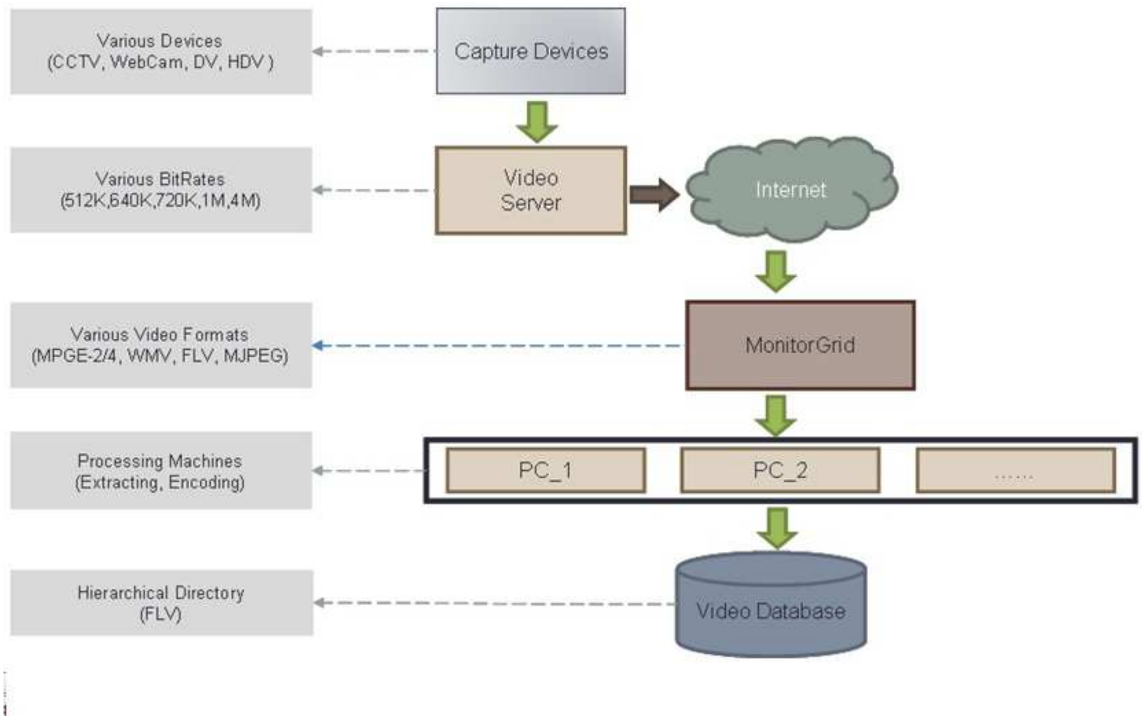

Figure 14: Process flow: from video capturing to storing

2.4.2 Storage and computing

One of the challenges to infrastructure building for the F4K project was the diverse requirements of different components. For example, the video processing components (detection, tracking, and recognition) needed fast computing facility, the database component needed fast I/O interface, and workflow and UI components need stable network which can transmit data flow seamlessly. To address the challenge we adopted the Infrastructure-as-a-Service (IaaS) model of cloud computing, storage and computing resources are consolidated in one single access framework. Figure 15 shows the conceptual architecture of infrastructure service framework. The three major components of the framework are: storage platform, computing platform, and service frontend.

Computing platform

In order to provide a flexible high-performance computing environment in support of the F4K project, we created a heterogeneous computing platform composed of a supercomputer system and a Virtual Machine cluster. Suitable middleware was developed to bridge two different computing systems so execution processes can automatically choose computing platform depends on the estimated execution time. Regarding the VM cluster with Grid engine system, a set of APIs were developed based on Grid Engine's DRMAA (Distributed Resource Management

Application) APIs to allow the workflow component to schedule job submission to the VM cluster. Access to the supercomputing system uses the LSF queuing system. A communication channel was created based on the SSH model between the VM master node and WindRider. Further, LSF APIs were created for process manipulations on WindRider through the SSH channel.

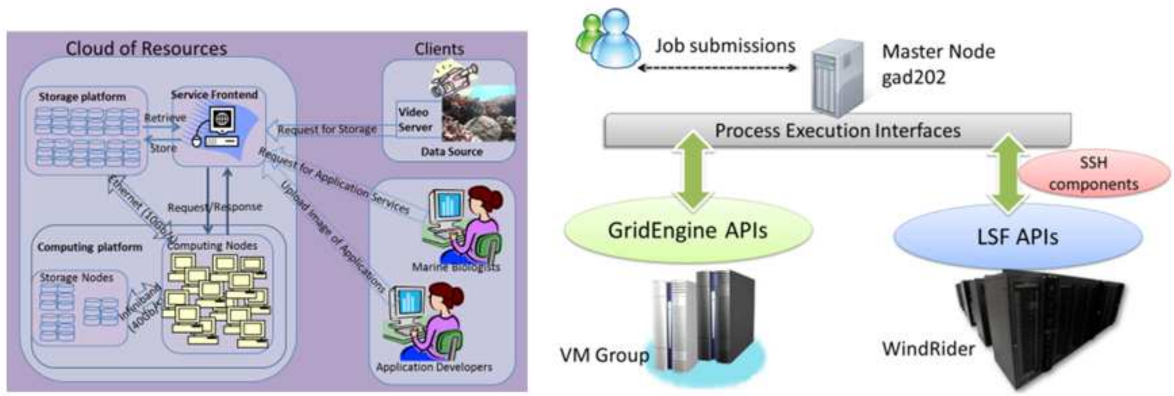

Figure 15: Architecture of infrastructure service. Left: conceptual architecture, right: the middleware, Job Dispatcher, bridges heterogeneous computing system.

Storage platform

As part of the infrastructure framework, we created a massive storage platform with capacity up to 210TB to fulfill the demand for longterm storage of the video data, both raw video and data after abstraction by the video processing components. A 10G network interface is used to connect the storage platform with computing platform and service frontend. The interconnection of components with broadband network assures data staging from storage platform to computing platform smoothly.

2.4.3 Database performance

The F4K SQL database is a collaborative effort between several teams that store shared results and information that can be accessed by everyone. The database was initially hosted by the University of Catania team in the drafting phase of first year. As the system evolved computing processes needed to communicate with the SQL database constantly and it was not effective to communicate through a long haul network. The database was moved to the NARL machine in the second year. Moving data closer to the processes allowed all the teams to achieve their desired goals more effectively.

In our design of the infrastructure service, computing intensive processes are distributed to machine cluster and running in parallel. While in the intensive processing phase massive, e.g. up to 1000 detection, tracking, and recognition tasks are running at same time and these processes need to communicate with the database instantly to retrieve parameters and store results. Rapid accesses to the database at same time created heavy loading on the database server which eventually became a bottleneck in the overall workflow. In the beginning of year 3, we identified two major bottlenecks of data communication, one is the network latency and the other is disk I/O latency. We also found the summary table aggregation process required dedicated resources, otherwise it dramatically slowed down other store/retrieve processes.

To resolve the disk I/O latency issues we adopted a load balancing mechanism which uses a two-node master-slave replication cluster and redirected read-write queries to the master and slave separately. Aggregation of summary tables was done once a week by the read-only slave. We also linked the system to SAN storage array with 4Gbs FC interface and have the database store on the SAN disk. We gained 1GB/s write, and 192MB/s read performance result from the move, and dramatically boosted efficiency of the detection processes that writes results into database heavily. The network latency was resolved by rerouting the database host to the same subnet as the computing platforms. Overall, we are able to accommodate more than a thousand processes communicating to the database server at same time. Table 7 summarized the main major SQL database tables and their physical size stored on disk.

2.4.4 Discussion

As the system evolved we learned some experiences which can benefit other researchers who are interested in implementing a similar framework. Details are given below:

1. The video processing tasks are classed as an 'embarrassingly parallel' workload which means tasks are mostly runnable independently. In this case, a 'semi-parallel' strategy was adopted which means we did parallelization at the shell script level. However, this strategy caused problems when we submit a job with more than 24 tasks in parallel. We found that a single task thread could have much less than 100% CPU usage and processes would slow down dramatically. That was because the scheduling policy on the supercomputer platform, Windrider, does not allow tasks to be distributed across nodes unless it was an MPI workload. There will be need to be an MPI distribution for feature work to harvest full computing power of computing platforms.

2. In the shared database access environment chances are tasks will read/write to the same table at same time. In general, the database server is smart enough to schedule workloads to prevent deadlock and return results instantly. However, sometimes, processes were implemented with an internal locking mechanism to prevent asynchronization of the read/write workload. In that case, internal table locking can cause deadlock to the system. We encountered several such kinds of deadlock to the system which did affect the progress of other tasks, because there is no other way but to reboot the server to resolve the problem. In any future implementation, we strongly suggest not to use internal locking mechanisms. The asynchronization problem can be resolved by presetting of initial parameters.

Table Name	Row count	Physical Size	Note
fish_detection	1445.41M	322.26G	Abstracted information for each detected object
fish_species	663.93M	24.67G	Correlated of fish object to species catalog
fish	124.28M	21.01G	Abstracted information of tracked fish objects
traj_species	97.29M	3.58G	Correlated tracking trajectory to species catalog
frame_class	11.61M	2.65G	Classification of video quality detailed to frames
fish_species_cert	32.55M	1.29G	Summary of detection/recognition certainty
summary_camera_39	7.13M	1.24G	Aggregation of information on camera id
summary_camera_46	7.12M	1.24G	
summary_camera_38	6.31M	1.10G	
summary_camera_37	4.46M	0.78G	
summary_camera_42	4.31M	0.75G	
summary_camera_44	1.49M	0.26G	
summary_camera_43	0.83M	0.15G	
video	0.63M	0.14G	Records of raw videos
processed_videos	0.78M	0.12G	Records of progress of video processing
summary_camera_41	0.63M	0.11G	
summary_camera_40	0.28M	0.05G	
video_class	0.53M	0.04G	Classification of video quality

Table 7: This table show the size of the current database hosted by NCHC (Taiwan), where the largest table is the fish_detection table. The first column indicates how many records (in millions are presented in the tables, the second column show the amount of raw data in Gb, while the final column give the amount of data that is really necessary for storage in the database because of indexing allowing also quick querying of this information.

2.5 WP 5: System Integration and Evaluation

2.5.1 T5.1 - Define component interfaces

Software Components

The Fish4Knowledge projects has a simple but effective design (described in Deliverable 5.1) which allows research from multiple field to make software which can easily interact with each other. The main idea is to communicate by means of the storage facility(s). This means that the data that is processed by the software components is available to all partners in the project, but more importantly to the end-user. The idea is that all components write their output to a storage facility. There will be a component (database component) that will collect and store the data, but also allows us to query and retrieve the data again.

We give an overview of how the components interact with each other by means of the storage facilities in the system. The videos from the underwater webcams are stored in the storage facilities, the Fish Detection/Tracking component will get the videos out of the storage facilities and will find the fish and label their locations in the frames (fish location) and follow fish in multiple frames. The Fish Detection/Tracking components will again store the obtained information (for example the fish locations) in the storage facility. The Fish Recognition component will try to determine the exact species label based on the stored fish locations and will store this as well. The User Interface is able to retrieve all the information previously stored in the storage facilities, and for instance count the number of species X during the month December. This is represented to the users, where the user interface also allows users to search through all the information in the storage facilities. The workflow component will check which system resources are available to process different Video/Image Processing components on new videos. For instance, it will keep track of the videos which have not been processed yet by the Fish Detection/Tracking component and the fish which have not been processed yet by the Fish Description/Recognition component. Furthermore, it should be able to handle special requests by the user interface, running different Video/Image Processing components.

In Figure 16, we show a schematic representation (UML Component Diagram) of the entire system. The components basically have interfaces and sockets and the information flow is given by the arrows. This schematic gives a rough overview, notice that this UML Component Diagram was developed at the start of the project and is still valid in most of the cases. The only changes are that some component, i.e the “Fish Description”- “Fish Recognition” and “Query Engine”-“User Interface” have become one component for practical reasons.

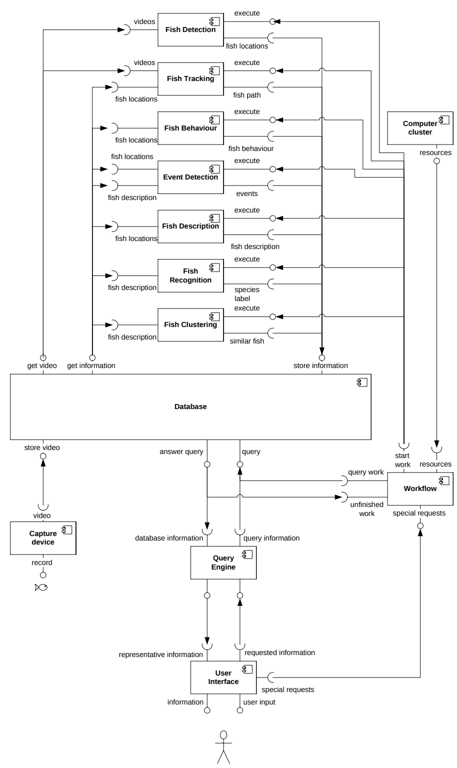

Figure 16: UML Component Diagram, showing the input and output relations of the different components

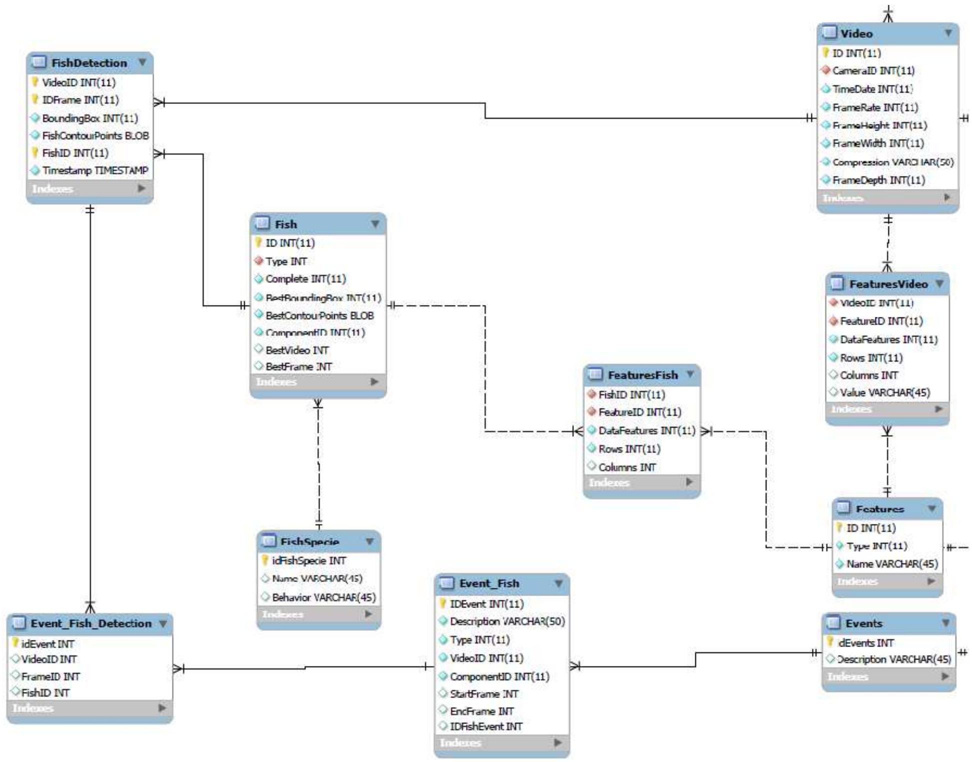

Figure 17: Part of the RDMS database schematic of the project showing the most important tables for the fish detection/tracking and fish recognition components

Database Definitions

The database definitions (defined in Deliverable 5.2) are an important part of the component interfaces, because information stored by a certain software component had to be read by another components. The database definitions were determined at the start (6 month) of the project and have only changed a couple of times slightly for practical reasons (limited storage capabilities). The RDF/RDMS Datastore Definition allow storing information on 1) the underwater monitoring system and on 2) the processing results in terms of fish detection, fish tracking, fish recognition, event detection and recognition, 3) processing of videos by the software components. The Software components inter-operate mainly by reading and writing data to a relational database conforming to the schema defined in Deliverable 5.2. Part of the database schema is shown in Figure 17. In addition, an RDF schema was defined in order to expose the project data in a Linked Data-compliant solution for Web-scale sharing of resources and experimental data as proposed in WP5.

2.5.2 T5.2 - Integration and evaluation planning

A timetable for the integration and evaluation was given in Deliverable 5.1. Although not all initial milestones were reached, already in the first year a database was setup that allowed the different groups in the project to share and reuse information. This database was first hosted in Catania and was later moved to Taiwan (second year), so that all groups were able to integrate their components with each other. By the end of the second year, the project showed during the second year review meeting a first working prototype of the entire system. This prototype was improved and extended during the final year based on an evaluation session with marine biologists and ecologists in Taiwan (Technical Project meeting in April 2013). Currently, one of the latest version of the user interface was presented at the European Marine Biologist Symposium 2013 in Galway, Ireland. Some new improvements have be made since this presentation.

Groundtruth Collection

For the evaluation, one of the challenges from the video analysis perspective was obtaining groundtruth annotations to verify the performance of the video analysis components. Multiple groundtruth annotation interfaces were developed in order to obtain data that allows us to evaluate the video analysis software, where the User Interface team (CWI) has been involved as well. Without this data, the evaluation of the components is impossible, but in most cases obtaining good quality annotations is difficult. In the Fish4Knowledge project, multiple tasks in video analysis like fish detection, fish recognition and behaviour classification needed to be evaluated using Groundtruth Image Data. These different classification tasks however also require different kind of interfaces for annotating the required Groundtruth data.

The main achievements in the area of ground-truth are:

1. Novel methods to enable non-expert users to perform expert image labelling tasks at a level comparable to that of experts.
2. Novel user interfaces for fish labelling for both expert and non-expert users.
3. Thousands of ground truth data collected and used for training the automatic recognition algorithms.
4. A rule-based framework for collecting and labelling ground-truth examples of fish behaviours.

A summary of the different interfaces for annotating the data is here (a more detailed description can be found in Deliverable 5.6):

1. Perla (fish detection): This is a web interface for labeling the contour and trajectory of fish in the videos. An example of this web interface is shown at the top of Figure 18. It allows multiple people to annotate the trajectory and the contour of the fish and later combine those annotations.

2. Flash the Fish Game (fish detection): The fish game (middle-left of Figure 18) is a fun way to perform the annotation of fish, where the annotator plays a diver in the game with a camera that has to take pictures of the fish. These picture allow us to define the location of the fish in the video. Notice however that these annotations do not give a contour.
3. Fish behaviour (fish behaviour): For the fish behaviour, an annotation website (middle-right of Figure 18) is created which allows users to search for combinations of species in the videos, for instance if two clown fish appear in the video around the same time. Afterwards, we can annotate if these fish are interacting with each other in certain way, for instance pairing.
4. Clustering interface (fish recognition): A website (bottom-left of Figure 18) was created to annotate the fish species, where we first remove the species that are incorrectly classified for that cluster and afterwards link the cluster to a certain species. This allows users to annotate fish images $3\times$ faster than annotating each image separately. It even makes the annotation task simpler as no domain knowledge is required.
5. Fish labeling game (fish recognition): This interface (bottom-right of Figure 18) transforms the difficult task of recognising fish species into an easier game task that only requires visual similarity judgements.

Based on our experience in creating datasets for new domain specific problems and dealing with having to evaluation large databases with noisy data with different kind of variance, we discovered that these are open and important problems which require more research. The Fish4Knowledge project organised two scientific workshops related to this problem (VIGTA 2012/2013), both workshops also led to organising a special journal issue on the same subject. (Special Issue on “Methods and Tools for Ground Truth Collection in Multimedia Application” of Multimedia Tools and Applications and Special Issue on “Large Scale Data-Driven Evaluation in Computer Vision” of the Computer Vision and Image Understanding Journal (Elsevier)). Given the current trends towards big data in research, groundtruth and the resulting evaluation based on the groundtruth will be issue that becomes more important for the scientific community.

2.5.3 T5.3 - First integration and evaluation phase

First prototype system

The first prototype was already an entire working system, where only a small number of issues needed to be resolved given the initial design.

The first prototype could process video and image data with multiple software components that can perform either fish detection and tracking or fish recognition. For fish detection, we had several different background subtraction methods and different fish tracking methods which could be used for this task. Two versions of the fish recognition software were available where the first version was able to recognize 10 species, and the newer version recognized 15 species.

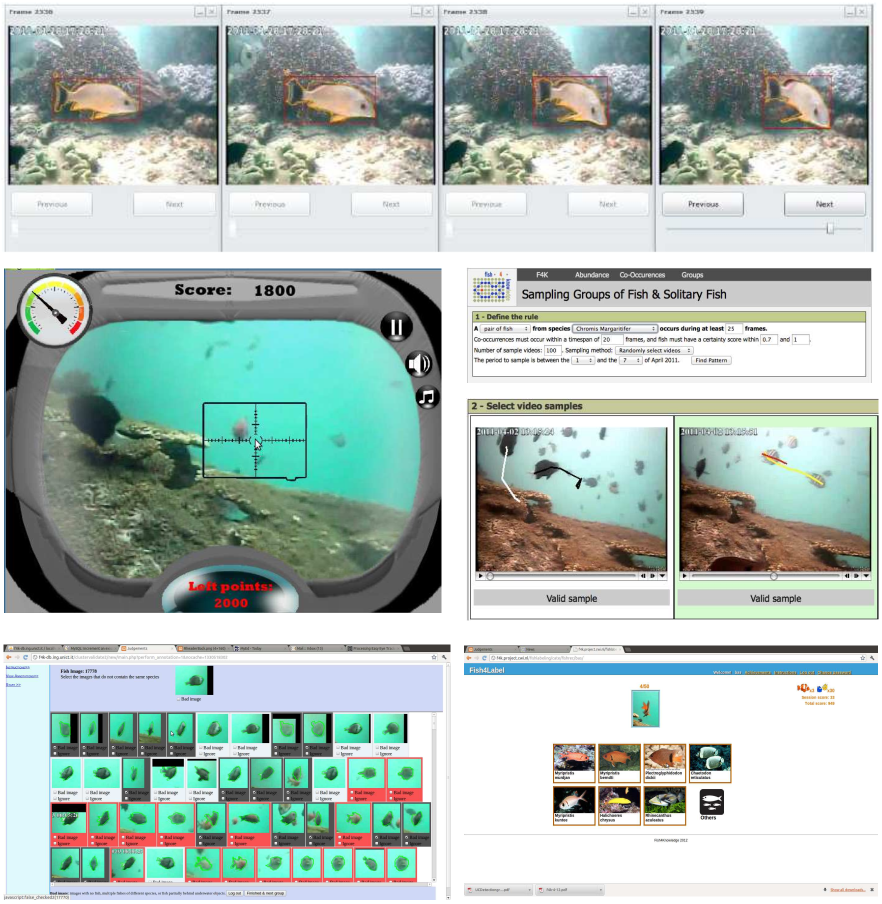

Figure 18: Examples of interfaces that have been developed for annotation of image processing groundtruth data

A simple bulk processing workflow was used to compute the backlog of video data with both the default fish detection and recognition software components. The user interface was able to show statistical information about the processed video data, where we had already years worth of processed video data stored in this system. More details can be found in in Deliverable 5.4 In the first prototype system, the bulk processing workflow needed to be replaced by a workflow that can perform bulk processing as well as response to user requests allowing users to run different versions of the software to verify for instance hypotheses. Here a connection between user interface and workflow was also still necessary.

As of May 27th 2013, the fish detection had processed 70784 clips of 10 minutes which was equal to around 983 days of video given the 12 daylight hours we are recording. The fish recognition, which depends on the fish detection component had processed around 67468 clips of 10 minutes (937 days of video). In total, we have however 623472 clips, although there are multiple clips where we have both low resolution and high resolution videos of the same scene (resulting in 528624 unique clips).

Measurements	Total	Fish Detection (May 2013)	Fish Detection (Oct 2013)	Fish Recognition (May 2013)	Fish Recognition (Oct 2013)
Processed Videos	528624	70784 (13.4%)	528624 (100%)	67468 (12.8%)	243563 (46.0%)
Processed Videos (class normal)	74611		74611 (100%)		75806 (98.42%)
Fish Fish Detections	124m 1445m			124m 1445m	53m 654m
Speed		40 min (std 83 min)	12 min (std 12 min)	175 min (std 381 min)	160 min (std 246 min)

Table 8: In this table, the comparison between the status at May 2013 of the processing videos and current status (October 2013), where large difference in the number of processed videos are shown. In the previous report, we already stated that lots of videos are blurred or have video encoding effect. Currently, we have method to filter out these kind of videos allowing us to focus first on the more promising class of normal videos

2.5.4 T5.4 - Second refinement and evaluation phase

Latest prototype system

For the first prototype, we reported that some components were not fully connected with each other (i.e. workflow and interface). This connections has been achieved allowing marine biologists to run video processing components on the videos in the database. Large improvements in the individual components are for the fish detection component that it is able to classify videos into categories like “blurred”, “normal”, “encoding problem”, etc. By looking at the information in the database, we discovered strange results, which by checking the original video where due to for instance “encoding problems” where often more fish are detected then one would normal expect. In the fish recognition components, we are able to recognise more species, going from 15 to 23 species. Also, the recognition component can filter out false positive from the detection stage. The user interface is improved in both the usability and the fact that it can present more views on the data. There is also a connection to the workflow giving marine biologists the ability to process videos with other VIP software. The detail about the improvements of the different components can be found in Deliverable 5.5.

Data processing status

The video and image processing modules analyse the video data detecting and recognising the fish in the video footage. The video data is saved in 10 minute video clips, where in total we have 528624 distinct video clips. For clarity reasons, we will state the both the number given in Deliverable 5.4 (which are measured at May 27th 2013) and the new numbers currently in the database in Table 8.

Table 8 shows that the fish detection component finished processing all the video in the database. The fish recognition component finished processing almost half of all the video, currently because of new filters we are focussing on the more promising class of normal videos (i.e. videos without blur or video encoding error), afterwards continuing on the rest of the videos. In Table 7, the largest tables in our database are shown, where both the number of rows (in millions) and the data size (in Gb) are shown for each table. The "fish_detection" table is the largest table. Other important tables are the "fish_species" where all the species information is storage, the "fish" table that contains the fish trajectory information and the "summary_camera_XX" tables that allow fast querying of the database. Given both Table 8 and Table 7, we show that this project is first of all truly a big data project. It is also on of the first projects able to analyse large amount of video data and to present the analysis to the users.

Discussion of Software Architecture

The Software Architecture proposed in this project, where the software components interface with each other using a database schema, allowed us to develop the individual software quickly without having to rely on the other teams in the beginning of the project. Also input from other teams could be easily tested when their results were stored in the database. The challenges at the start however involved getting the database definition stable (which took some time also because of storage capacity issues). Another problem is that often new definition or functionalities are necessary, however not all partners might be aware of the function of new field in the database. The advantage is that developers can easily add new field in the database, without other developers having to redesign the software. The disadvantage is that persons need to be informed about the field definitions once they become widely used, which this project achieved mainly by a mailing list and updating Deliverable 5.2. This architecture is ideal for lots of scientific big data projects, where it allows a lot of freedom to the software engineers and software can easily contribute by filling the database with new processed data.

Discussion on Future Improvements

Although is project achieved both in the processing of video data and the collection of groundtruth data outstanding results. We discovered that linking the groundtruth data to the processed video data is difficult and might be a nice subject for future projects. The idea that groundtruth data, which always comes from relative small subsets of the original data, should be representative for the entire dataset, allows to give marine biologist a feeling for the performance of the system. This is basically what we achieved within this project. However, it might lead to some kind of automatic manner to normalize the observation by automatic methods. We also discovered that the entire dataset probably contains subsets which makes the original assumption that the groundtruth data is representative for the entire dataset false. Subsets of the groundtruth data can also be generated (i.e. in our data "normal"."blurred","video encoding error") in which case we have to link to subsets that represent the data to normalize for certain effects. Investigation of future project can link ground truth and observations in a more fundamental way, where also the underlying storage architecture has to support this.

3 Project Dissemination and Impact

As a part of the F4K project, we have promised to develop a project web site including on-line data repositories, organise 4 workshops, develop public data exploration interfaces, and promote the methodology to the marine biology community. Over the span of F4K's three project years, we have fulfilled the above project promises and much more. A fully fledged project web site ladened with data archives, ground truth, source code, open access versions of published papers and report publications is publicly available at:
`http://groups.inf.ed.ac.uk/f4k/resources.htm`.

Six publicly accessible user interfaces are available through seven web sites/portals. These are the public user interface for using the overall F4K system; or public access for participating our crowd sourcing projects in an attempt to evaluate our video processing modules. Some of such efforts are appeared in an on-line game format. In addition, the F4K Underwater Aquarium is hosted through an on-line 3D virtual world environment in Second Life.

Overall, we organised 6 scientific workshops and organised 5 special issues for publishing in scientific journals. During the last year of the project, we started to write an overall F4K project book in preparation for publishing in 2014.

3.1 The F4K Project Book - A F4K Legacy

The F4K book will present an integrated, inter-disciplinary, computational approach to the capturing, analyzing, interpreting, presenting and managing of mass volumes of marine video data that has been captured from the open sea. It will provide its readers a rare opportunity to gain an overview of a set of relevant, supportive, inter-disciplinary and innovative research and technical works in one place. This gives a coherent view of the research papers published in different scientific domains. It would be a useful reference book for researchers and practitioners who are interested in handling big data that may be gathered from the Web or the natural world, as it provides ene-to-end detailed descriptions and insights as how these complex tasks have been accomplished.

The F4K project book timely deals with big data research and innovation issues and will be a great F4K project legacy. We have received high praises from book proposal reviewers and publisher and are awaiting a book contract from Springer to be published in the "Intelligent Systems Reference Library" book series. We plan to submit a final draft next spring for publishing in 2014.

3.2 F4K Led Special Issues in Scientific Journals

- The special Issue **"Large Scale Data-Driven Evaluation in Computer Vision"** of Computer Vision and Image Understanding Journal (Elsevier) aims at presenting and reporting

the most recent efforts: 1) to support automatic or semi-automatic generation of large scale datasets together with annotations, 2) to integrate existing datasets by investigating harvesting and representation schema matching approaches, 3) to exploit big visual data and the Internet crowd to overcome the lack of annotated datasets and 4) to develop 'data-driven' approaches also able to evaluate algorithms' performance with limited or no ground truth data. The call for papers has already been circulated and the foreseen deadline for papers submission is Nov 30, 2013, while the expected publication date is September 2014.
Guest editors: Dr. C. Spampinato (University of Catania, Italy), Dr. B. Boom (University of Edinburgh, UK) and Prof. B. Huet (EURECOM, France)

- Special Issue **"How Can Multimedia Help Ecology?"** of the Multimedia Systems Journal (Springer). This special issue will present and report on the most recent methods for the management, processing, interpretation, and visualisation of multimedia data recorded for monitoring ecological systems with aim to provide powerful tools to make ecologists understand and model different aspects of life: from interactions among small organisms to processes spanning the entire planet. The call for papers has already been circulated and the foreseen deadline for papers submission is December 31, 2013.
Guest editors: Dr. Concetto Spampinato (University of Catania, Italy), Dr. V. Mezaris (CERTH-ITI, Greece) and Dr. Jacco van Ossenbruggen (CWI, The Netherlands)

- **"Ground Truth Collection in Multimedia"** in the Multimedia Tools and Applications Journal (Spinger). This special issue addresses the development of: multimedia processing methods for supporting automatic ground truth generation, methods and tools for combining and comparing ground truth labeled by multiple users in any field of multimedia where ground truth is required, interfaces (adaptive, proactive, mobile, web-based) for collecting ground truth, methods for data representation and integration, interoperability middleware, features, algorithms, and tools. Guest editors are: Concetto Spampinato (University of Catania, Italy), Bas Boom (University of Edinburgh, UK), and Jiyin He (CWI, the Netherlands).

- **Methods and Tools for Ground Truth Collection in Multimedia** of Multimedia Tools and Applications Journal (Springer): The special issue specifically addresses the development of: multimedia processing methods for supporting automatic ground truth generation, methods and tools for combining and comparing ground truth labeled by multiple users in any field of multimedia where ground truth is required for collecting ground truth, methods for data representation and integration, interoperability middleware, features, algorithms, and tools. The CfP of the special issue was circulated to about 5000 researchers in multimedia processing and attracted 14 papers, from which 9 papers were accepted to appear in the special issue and their online version is already available. The topics of the accepted papers range from general purpose video annotation tools for image segmentation to approaches for supporting labeling of shadow, head pose and vehicle to benchmarking platforms for evaluating color texture classification schemes to requirements of metadata schema for performance evaluation. The guest editors of this special issue were Dr. Spampinato, Dr. Boom and Dr. He. The printed version of the special issue is expected to be published early 2014.

- **Multimedia in Ecology** of the Ecological Informatics Journal (Elsevier): The special issue ‘Multimedia in Ecology’ specifically reports on the most recent methods for the processing, interpretation, and visualization of multimedia data recorded for monitoring ecological systems, with particular attention to animal and plant identification and classification and pollution monitoring. The CfP of the special issue was circulated to about 3000 researchers in multimedia and ecoinformatics and 15 articles were received, from which 11 papers accepted to appear in the special issue. The topics of the accepted papers deal mainly with animal/plant identification and recognition, habitat classification and frameworks for sensing the environment and monitoring the pollution. The guest editors of this special issue were Dr. Spampinato, Dr. van Ossenbruggen, Dr. Huet and Dr. Mezaris and the special issue is expected to be published early 2014.
- **Animal and Insect Behaviour Understanding in Image Sequences** of EURASIP: This special issue aims at reporting on the recent approaches and tools for the identification, interpretation and description of animal and insect behaviour in image sequences. It specifically focuses on the interactions between (i) computer vision theories and methods, (ii) artificial intelligence techniques for the high-level analysis of animal and insect behaviours and (iii) multimedia semantics methods for indexing and retrieval of animal and insect behaviour detected in images and videos. The special issue was specifically designed to publish the best papers presented both at MAED’12 and at VAIB’12. However, a CfP was also circulated to about 10.000 researchers resulting in 14 submitted papers, from which eight have been accepted, four rejected and two are still under review. The topics of the submitted papers are in line with the ones called, in detail: 2D and 3D methods for automatic detection, tracking and recognition of animals (mice, elephants, chimpanzees) and insects (mainly bees) in ‘real-life’ environments by processing images, videos and audio. Approaches to investigate animal and terrestrial insects’ behavior both in real-life scenarios (e.g. stickleback schooling behavior) and in a lab setting were also submitted. The guest editors of this special issue were Dr. Spampinato, Dr. Boom, Dr. Farinella, Dr. Mezaris, Prof. Betke and Prof. Fisher and the special issue is expected to be published on June 2014.

3.3 F4K Led Scientific Workshops

- **The Intelligent Workflow, Cloud Computing and Systems workshop** as a part of the KES-AMSTA conference, Manchester, UK, June 29-July 1, 2011. It was co-organised by Dr. Yun-Heh Chen-Burger (University of Edinburgh, UK), Prof. Ching-Long Yeh (Tatung University, Taiwan), and Dr. Fang-Pang Lin (NCHC, Taiwan). There was 1 invited and 4 submitted talks. Around 20 people attended the workshop.
- **The Intelligent Workflow, Cloud Computing and Systems workshop** as a part of the

KES-AMSTA conference, Dubrovnik, Croatia, June 2012. This workshop is in its 3rd year running. It is co-organised by Dr. Yun-Heh Chen-Burger (University of Edinburgh, UK), Prof. Ching-Long Yeh (Tatung University, Taiwan), Prof. Lakhmi Jain (University of South Australia, Australia), and Dr. Fang-Pang Lin (NCHC, Taiwan). There were five paper presentations included in the conference proceeding and to be included in Springer's LNAI series. Around 30 people attended the workshop, many of them are returned participants.

- The **International Workshop on Video and Image Ground Truth computer vision Applications (VIGTA'12, VIGTA'13)**. These were held in conjunction with the Advanced Visual Interfaces (AVI 2012) International Conference in Capri Italy, May 21-25, 2012 and in St. Petersburg (Russia), July, 2013 in conjunction with the International Conference on Computer Vision Systems (ICVS 2013). The workshop aimed at reporting on the most recent methods to support automatic or semi-automatic ground truth annotation and labelling as well as algorithms' performance evaluation and comparison in many applications such as object detection, object recognition, scene segmentation and face recognition both in still images and in videos.

  The call for papers attracted 15 papers from which 8 were selected for oral presentations. The topics of the accepted papers range from how to use internet images and semantic web technologies for supporting image and video annotation to the generation of large scale ground truth by reporting to crowdsourcing to performance evaluation and comparison of computer vision methods.

  The workshop featured two keynote talks: "Overview of Quality Assessment for Visual Signals and Newly Emerged Trends" given by Prof. Ngan, King Ngi from Chinese University of Hong Kong; and "Ground Truth Design Principles – An Overview" given by Dr. Kondermann from Heidelberg Collaboratory for Image Processing (HCI) of University of Heidelberg. 35 people attended the workshop.

  The workshop proceedings are published by the ACM International Conference Proceeding Series published by ACM. The workshop chairs were: Dr. Concetto Spampinato (University of Catania, Italy), Dr. Bas Boom (University of Edinburgh, UK) and Prof. Benoit Huet (EURECOM, France).

- The **Second ACM International Workshop on Multimedia Analysis for Ecological (MAED'13)** in Barcelona (Spain) on October 21, 2013 in conjunction with the ACM Multimedia Conference, aimed at bringing together a cross-disciplinary crowd of people in order to investigate current and emerging topics within ecological multimedia data analysis. The workshop, in particular, outlined the state of the research on the most recent methods for the processing and interpretation of multimedia data recorded for monitoring ecological systems.

  In total, the Program Committee accepted 7 papers (from 12 submitted papers) covering the following topics: Animal detection and recognition by processing image, video and audio data; fish and marine environment monitoring; benchmarking and user-appreciation of ecological multimedia technologies and applications in biogeography.

The workshop also features two keynote talks: 1) ”Collection and Analysis of Two Complex Ecological Datasets” given by Prof. Robert Fisher from the School of Informatics, University of Edinburgh, and 2) ”Understanding Animal Flight with Three-dimensional and Infrared Computer Vision” delivered by Prof. Margrit Betke from Boston University. The workshop proceedings are published by the ACM International Conference Proceeding Series. The workshop chairs were: Dr. Concetto Spampinato (University of Catania, Italy), Dr. Vasileios Mezaris (CERTH-ITI, Greece), Dr. Jacco van Ossenbruggen (CWI, The Netherlands).

- The special session on **Image Processing and Pattern Recognition for Ecological Applications** was organised as part of the 2013 IEEE International Conference on Image Processing[2], Melbourne, Australia, September 15^{th}-18^{th}, 2013. This special session, mainly, reports on the most recent pattern recognition approaches in several fields of ecology from plant recognition to natural habitat classification to animal behaviour understanding. The special session organisers were Dr. Concetto Spampinato (University of Catania, Italy), Dr. Vasileios Mezaris (CERTH, Greece) and Dr. Alexis Joly (INRIA, France).
- The Fish4Knowledge team organised a one day workshop on “Visual observation and analysis of animal and insect behavior”, held on November 11, 2012, as part of the 21th Int. Conference on Pattern Recognition (ICPR), Tsukuba, Japan. The workshop organisers were R. Fisher (University of Edinburgh), J. Hallam (University of South Denmark), and B. Boom (University of Edinburgh). 24 extended abstracts were received and each was reviewed by 3 members of the organisers and programme committee, from which 18 talks were accepted. About 35 people attended the workshop. See: `http://homepages.inf.ed.ac.uk/rbf/vaib12.html` for more details and papers.
- **Live F4K UI demonstration workshops**: three were held in Taiwan targeting (marine) biologists/ecologists, including researchers in coral reef fish, corals, plankton, microorganisms and ecotoxicology. The demonstrations were held at 2 laboratories of Academia Sinica (Systematics and Biodiversity Information, Taipei, and ICOB, Yilan), and at the National Museum of Marine Biology and Aquarium (Kenting). More than 30 participants attended the workshops.

  During the workshops, participants were shown a presentation about the F4K project, explaining the means to detect fish, recognise fish species and their behaviors in video footage. We also provide interpretations on the video footage analysis and methods to use the F4K user interface to acquire desirable results. For the Kenting workshop, participants simultaneously interacted with the F4K system using 20 computers. We have collected feedback from participants, inc. desirable use of the F4K system for scientific research and possible refinements. We have also recruited participants for our user study, details

[2] www.ieeeicip.org

are provided in deliverable D6.6.

3.4 Invited Talks, Posters and Exhibitions at Scientific Conferences

- Invited talk on **Development of Earth Science Observational Data Infrastructure of Taiwan, inc. Introduction of Fish4Knowledge project** GLIF (Global Lambda Integrated Facility), Oct 4, 2013, Singapore.
- Presentation on F4K development in the Telescience Working Group of PRAGMA, Pacific Rim Applications and Grid Middleware Assembly, March 20-22, 2013, Bangkok, Thailand.
- Presentation on F4K results in the Telescience working group of PRAGMA, Pacific Rim Applications and Grid Middleware Assembly, October 16-18, 2013, Beijing, China.
- Invited talk on **Business Process Modelling, Logic, Semantics based Reasoning and Intelligent Workflow** at the Bridging Big Data Infrastructures - Expedition on the Network Science Landscape workshop, December 3-6, 2012, NCHC, Taichung, Taiwan. This talk gave the linkage between BPM, intelligent workflow and virtual workflow machines; and how logic and semantic reasoning played an important part in providing run-time (re-)configurable virtual workflow machine.
- Invited talk on **Workflow Management and Fault Tolerance** at NCHC, Taiwan, March, 2013. This talk presented the workflow engine, error detector and repair. It gave insight on how the F4K data and HPC facilities were utilised on an on-demand basis. Particularly, we discussed the decision-making strategies from the workflow performance's point of view and considered factors from NCHC's database and HPC specialisation.
- A poster describing the F4K project was presented at the **21st European Association of Fisheries Economists (EAFE) 2013** conference, held at Heriot-Watt University in Edinburgh from 15-17 April. "The theme of the conference was: Securing the future - Implementing reform in European Fisheries. Keynote speakers included Ms Lowri Evans, Director General of DG Mare, Mr Richard Lochhead MSP, Cabinet Secretary for Environment and Rural Affairs, Prof Thomas Sterner, visiting Chief Economist at EDF, Prof Ragnar Tveteras, Head of Stavanger Centre for Innovation Research and Mr Mike Park, Chief Executive of the Scottish White Fish Producers Association. Over 90 delegates from around Europe and the world attended from places as far afield as Japan, Alaska and throughout Europe."

  While the focus of the event was on wild and farmed commercial fishing, we thought that they would be interested in the fish detection and recognition technology. We had a poster that summarised the project, which was seen by many delegates, of whom 4 spoke with us in more depth.
- UEDIN will participate in a poster presentation at the Scottish Informatics and Computer

Science Alliance (SICSA) in association with Scotland IS, who are hosting their 6th annual DEMOfest. This is a technology showcase that brings together industry and academia and get to meet others in the SICSA research community. This year it will take place on 5th November 2013 in the Mitchell Library, North Street, Glasgow, G3 7DN. Based on previous years, we estimate that 100+ delegates will see the poster.

- Prof. Fisher will give a keynote talk on "Applying Computer Vision Methods to Ecological Problems", at the 2013 IEEE Second Int. Conf. on Image Information Processing (ICIIP -2013), December 9 - 11, 2013, Jaypee University of Information Technology, Shimla, Himachal Pradesh, INDIA
- At the **48th Annual European Marine Biologist Symposium (EMBS 2013)**, a poster was presented together with a youtube movie `http://groups.inf.ed.ac.uk/f4k/`.

  This was shown during the introduction speech at the symposium. This symposium was held at the National University of Ireland, Galway. During the poster presentation, we showed marine biologists/ecologists our system which we demonstrated live at the symposium. The youtube movie which was posted before the symposium already attracted some of the participants to try out the system before the symposium. The symposium had 85 oral and 85 poster presentations, with an International audience coming for mostly Europe, but also the United States, Australia, Russia, China.
- At the **9th Indo-Pacific Fish Conference (Okinawa)**, a talk about the F4K system was presented, with particular focus on the user interface since the this is the user community. People weren't at all interested in our data (since it wasn't collected to answer their questions), but they are interested in how to re-use or adapt the algorithms for their own videos.
- Exhibition on **Data Infrastructure for Fish4Knowledge**, International Supercomputing Conference, 17-19, June, Leipzig, Germany.
- Joint exhibition on **Demo of Fish4Knowledge and with Wailalak University, Thailand on live video streaming from Racha Yi Island to Denver** Supercomputing Conference, 17-22, November, Denver, Colorado, USA.
- A keynote talk on "Experts, non-experts, and automatic methods in crowdsourcing in wildlife image annotation" at the 1st Int. workshop on Social Media for Human Computation, in conjunction with the IEEE social computing conference, 2012, Amsterdam. The talk discussed issues in and approaches to crowdsourcing tasks that require specialists' knowledge based on our studies in the F4K project.
- A talk on "Comprehensive visualization of underwater video data: uncertainty, provenance and multidimensional analysis in the Fish4Knowledge project" and a demo at the workshop Large Data Analysis in Marine Biology Science: New Possibilities through Visual Analytics, in conjunction with the 9th Baltic Sea Science Congress, on August 28th 2013. The audience was composed of researchers from both the marine biology and the HCI domains, raising awareness of the project in both communities.

- A demo at the launch event of the Data Science Research Center of the University of Amsterdam, on November 13th 2013. The audience is mainly composed of computer scientists from a wide range of domains (e.g., Information Retrieval, Machine Learning, Database, High-Preformance Computing), as well as scientists, such as astronomers, applying the tools developed by the computer scientists.

3.5 Fostered Collaboration

- New scientific proposal by Dr. Fang-Pang Lin, NCHC, jointly submitted with marine biology researchers from National Museum of Natural Science and Aquarium to NSC, Taiwan.
- Collaboration with University of Cardiff, UK, and Universidad de Zaragoza, Spain, on workflow performance analysis and fault tolerance: this has led to a co-authored published conference paper in 2013. Dr. Jessica Chen-Burger and colleagues are currently preparing for a journal paper and an EU proposal for Horizon 2020.
- Organizing a video-based fish identification task within the lab LifeCLEF 2014 (part of the ImageCLEF initiative)
- Part of the organization of the Background Modeling Challenge 2014, which, unlike BMC 2012 , will include underwater videos.
- Collaboration with Prof. Margrit Betke from the Boston University on the generation of large-scale ground truth for object detection. A joint proposal between Prof. Giordano, Prof. Betke and Dr. Spampinato was submitted to the last Marie Curie Outgoing fellowship programme.
- The AQUACAM Research Programme is a 3-year collaboration (2012-2015) between the Fish4Knowledge Research Consortium (F4K), The University of the West Indies (UWI) and The CARIBSAVE Partnership. The goal is to develop a new monitoring system for tropical reef fish, using fixed underwater video cameras and computer vision software that can detect and recognize approximately 40 species of Caribbean fish and estimate their body length. Collaboration includes participation in the supervisory committee of a PhD student from UWI who is researching the methodological aspects of estimating fish biomass by using F4K technology as compared to other methods used by marine biologists. During F4K project year 3, the following activities have been done for the AQUACAM research programme:
  - Adaptation of the fish detection and tracking approaches developed within F4K to deal with higher spatial and temporal underwater videos taken in the Caribbean;
  - Labeling of 10K objects on the new videos for exhaustive performance evaluation;
  - Set up of the video server for collecting and the sharing the taken videos;

  – Study of the state of the art of the stereo approaches for fish size estimation.

- Discussions has started with Dr. Tung-Yung Fan, a marine biologist at the National Museum of Marine Biology and Aquarium, Taiwan, regarding future collaboration on coral reef observation and growth, well-being monitoring and potential disease discovery and recovery. Dr. Fan is a world-leading experts in this field. He also has extensive connections in the US and Australia research communities.
- NARL is now taking advantage of the F4K computational framework starting from October 2013 to develop a peta-scale Earth Science Observational Knowledge base, which incorporate data collected from Taiwan's remote sensing and meteorological satellites, ocean research ships, meteorological and geological ground stations, and processes of domain specific analytics into a single service system.
- Organisation of the background modeling challenge 2014 (likely in conjunction with the ECCV'14 conference) with Prof. Vacavant (University of Auvergne, France). Next year the challenge will also include underwater video sequences.
- Collaboration with Dr. Daniel Kondermann from the Heidelberg Collaboratory for Image Processing (HCI) group to build up a European working group for performance evaluation in computer vision.

3.6 Publicly Available Resources

To aid the computer vision and marine ecology research communities, we have made two subsets of the raw and processed video data available: 1) All Years: a 10 minute video clip from all cameras taken at 08:00 every day in the project Oct 1, 2010 - July 31, 2013, giving approximately 10K video clips. This data allows analysis of fish patterns over annual cycles and comparison between sites. 2) Full Day: all 720 video clips from the 10 cameras taken from 06:00 - 18:00 on April 22, 2011. This data allows analysis of fish patterns over a full day period and comparison between sites. The SQL associated with each video clip is also made public. The data can be found at:

```
http://groups.inf.ed.ac.uk/f4k/F4KDATASAMPLES/...
    ...INTERFACE/DATASAMPLES/search.php
```

In addition, we have made several user interfaces publicly available. A list of such UI is provided below:

- **Main Interface to the Fish4Knowledge System:**

  Asia: `http://gleoncentral.nchc.org.tw`

  Europe: `http://f4k.project.cwi.nl`

- **F4K fish detection, tracking, recognition, and behaviour ground truth:**

The raw data and groundtruth results for these 4 processes are at:

`http://groups.inf.ed.ac.uk/f4k/GROUNDTRUTH`

- **F4K Underwater Aquarium and Exhibition Hall:** this virtual building is available via the on-line virtual world environment in Second Life at:

  `http://maps.secondlife.com/secondlife/Edinburgh%20University/70/198/26`

- **Flash the Fish game:** Flash the Fish is an on-line crowd sourcing game that enables massive automated annotations on videos by gaining game points. The goal of the game is to spot/flash a fish in underwater videos, by clicking on them to gain as many points as possible.

  `http://f4k.dieei.unict.it/fish_game`

- **Fish Labelling Game for non-experts:** match the fish with the right species -

  `http://f4k.project.cwi.nl/fishlabeling/accounts/login/`

- **Automated Fish recognition sites for experts:**

  `http://f4k.project.cwi.nl/labeling/clusterlabels/`.

- **Cluster the Fish - a Crowd Sourcing UI:** for custering fish of the same species -

  `http://f4k.dieei.unict.it/fish_labeling`

References

[1] R. Tolosana-Calasanz, M. Lackovic, O. Rana, J. Banares, D. Talia, “Characterizing quality of resilience in scientific workflows”. In *6th workshop on Workflows in support of large-scale science*, pp. 117-126, 2011.

[2] Olivier Barnich and Marc Van Droogenbroeck. ViBe: a universal background subtraction algorithm for video sequences. *IEEE Transactions on Image processing*, 20(6):1709–1724, June 2011.

[3] J. Benois and D. Barba. Image segmentation by region-contour cooperation for image coding. *Proceedings., 11th IAPR International Conference on Pattern Recognition. Vol. III. Conference C: Image, Speech and Signal Analysis,*, 1992.

[4] Mihir Jain, Herve Jegou, and Patrick Bouthemy. Better Exploiting Motion for Better Action Recognition. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR ’13, pages 2555–2562, Washington, DC, USA, 2013. IEEE Computer Society.

[5] Kyungnam Kim Kyungnam Kim, T.H. Chalidabhongse, D. Harwood, and L. Davis. Background modeling and subtraction by codebook construction. *2004 International Conference on Image Processing, 2004. ICIP ’04.*, 5, 2004.

[6] Guang-Hai Liu and Jing-Yu Yang. Image retrieval based on the texton co-occurrence matrix. *Pattern Recognition*, 41(12):3521–3527, 2008.

[7] Fatih Porikli. Achieving real-time object detection and tracking under extreme conditions. *J. Real-Time Image Processing*, 1(1):33–40, 2006.

[8] Fatih Porikli, Oncel Tuzel, and Peter Meer. Covariance Tracking using Model Update Based on Lie Algebra. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[9] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances In Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[10] Y.-C. F. Wang and D. Casasent. A support vector hierarchical method for multi-class classification and rejection. In *International Joint Conference on Neural Networks, 2009. IJCNN 2009*, pages 3281–3288, 2009.

[11] E. Beauxis-Aussalet, E. Arslanova, L. Hardman, and J. van Ossenbruggen. A case study of trust issues in scientific video collections. In *Proceedings of the 2nd ACM international workshop on Multimedia Analysis for Ecological Data*. ACM, 2013.

[12] E. Beauxis-Aussalet and L. Hardman. Fish4knowledge deliverable D2.2 User Scenarios and Implementation Plan. `http://Fish4Knowledge.eu/DELIVERABLES/F4K_Del2-2_v3-9.pdf`.

[13] E. Beauxis-Aussalet, J. He, C. Spampinato, B. Boom, J. van Ossenbruggen, and L. Hardman. Fish4knowledge deliverable D2.3 Component-Based Prototypes and Evaluation Criteria. `http://groups.inf.ed.ac.uk/f4k/DELIVERABLES/F4KDel23.pdf`.

[14] E. Beauxis-Aussalet, S. Palazzo, G. Nadarajan, E. Arslanova, C. Spampinato, and L. Hardman. A video processing and data retrieval framework for fish population monitoring. In *Proceedings of the 2nd ACM international workshop on Multimedia Analysis for Ecological Data*. ACM, 2013.

[15] E. Beauxis-Aussalet, J. van Ossenbruggen, and L. Hardman. Fish4knowledge deliverable D2.1 User Information Needs. `http://Fish4Knowledge.eu/DELIVERABLES/Del21.pdf`.

[16] E. Beauxis-Aussalet, J. van Ossenbruggen, and L. Hardman. Fish4knowledge deliverable D2.3 - Appendix I - How UI features address biologists requirements. `http://groups.inf.ed.ac.uk/f4k/DELIVERABLES/D2.3-appendix.pdf`.

