

# Pattern Recognition for Ecological Science and Environmental Monitoring: An Initial Report

Eric N. Mortensen, Enrique L. Delgado, Hongli Deng, David Lytle, Andrew Moldenke, Robert Paasch, Linda Shapiro, Pengcheng Wu, Wei Zhang, Thomas G. Dietterich

## **Abstract**

Many ecological science and environmental monitoring problems can benefit from inexpensive, automated methods of counting insect and mesofaunal populations. Existing methods for obtaining population counts require expensive and tedious manual identification by human experts. This chapter describes the development of general-purpose pattern-recognition algorithms for identification and classification of insects and mesofauna and the design and construction of mechanical devices for handling and photographing specimens. This chapter presents techniques being explored in the first two years of a four year project, along with the results obtained thus far. This project's primary focus to date has been the classification of stonefly larvae for assessment of stream water quality. Imaging and specimen manipulation apparatus that semi-automatically provides high-resolution images of individual specimens from multiple angles has also been designed and assembled in the context of this project. An additional project target has been the development of robust classification algorithms based on interest operators, region descriptors, clustering, and 3D reconstruction to automatically classify each specimen from its images.

## **1 Introduction**

Progress in the ecological sciences is limited by the lack of high-resolution sensing of either the abiotic or the biotic biosphere. While remote sensing from satellites or aircraft provides valuable information about gross spatial distribution by organism type (e.g., broadleaf versus needle leaf trees), high-resolution measurement of organism population sizes and spatio-temporal distribution is virtually impossible to acquire. Existing methods for obtaining population counts involve the manual collection and identification of specimens by human experts, which is too costly to provide ongoing high-resolution data. Among the many technologies being developed to address this problem, pattern recognition from image data is one of the most promising.

This chapter presents the results to date in an ongoing project by a multi-disciplinary team of computer scientists, entomologists, and mechanical engineers to develop high-throughput methodologies for the identification and classification of insects. Mechanical devices for automatically photographing insect specimens have been developed along with general-purpose pattern recognition algorithms for

classifying these specimens to genus or species levels. These methods and devices are applied to two important scientific, environmental, and agricultural problems: (a) water quality monitoring in streams (by recognizing and counting stonefly larvae) and (b) measurement and characterization of soil biodiversity (by recognizing and counting soil mesofauna).

A fundamental scientific challenge for computer science research is to develop general-purpose pattern-recognition methods that can be applied to many different classification problems without requiring manual redesign for each new task. Some existing pattern-recognition methods in systematics require carefully designed feature extraction and/or classification algorithms for each new application [[14],[32],[35]]. Consequently, each new application requires substantial time and expertise to construct. Our work addresses this challenge by developing a robust pattern-recognition system that can be applied without modification to various situations.

A second fundamental challenge is to develop pattern-recognition methods that can handle highly-articulated 3D objects. Many existing pattern recognition methods are largely limited either to objects or object parts that are roughly 2D (e.g., insect wings) [[32],[35]] or to specific views of semi-rigid 3D object parts (e.g., human faces, spider genitalia, etc.) [37]. The insects studied in this project are 3D objects with many articulated parts (legs, antennae, abdomen, tails, etc.) that cannot be reliably placed into consistent poses. To address this challenge, we are applying recently-developed computer vision techniques that detect distinctive image regions, represent them in ways that capture important invariants that are then combined to classify the specimens.

## **2 Project Overview**

While the environmental monitoring tasks of identifying stonefly larvae and counting soil mesofauna populations have quite different characteristics, our goal is to develop a robust pattern recognition system that can adapt to new identification tasks simply by relearning each new domain from a set of training data. Here we present an overview of the two tasks and our approach to classification. Note that this overview discusses both completed and in-progress work.

Figure 1 illustrates the entire classification system from imaging of specimens to taxonomic identification. The stonefly larvae and soil mesofauna are prepared and mechanically manipulated for imaging using different methods and mechanical hardware. Likewise, the software control of the mechanical apparatus is, by necessity, performed by different modules within the integrated imaging software. Images captured by the digital camera are first segmented to identify the image regions belonging to specimen(s) and to separate the specimen(s) from the background. Each foreground region in the segmented images (ideally corresponding to individual specimens) is then categorized by means of a coarse classification that groups specimens using simple-to-compute object properties (e.g., eccentricity, color histograms, compactness, etc.). The segmented images and the specimen's coarse grouping are then employed for fine classification where the goal is to identify each specimen to the species level, though in some cases classification to genus or even just to family is beneficial. As this

project is still a work in progress, components of the system that are under development have dashed outlines.

The remainder of this section discusses the two application problems: recognizing stonefly larvae and classifying soil mesofauna. For each application, we describe the motivation for choosing the taxonomic identification task and the methods for manipulating specimens and capturing images.

## **2.1 Recognizing Stonefly Larvae**

Stream water quality measurement could be revolutionized if an economically-practical method were available for monitoring aquatic insect populations. Since species differ in their water quality requirements, population counts of stonefly (Plecoptera) larvae and other aquatic insects inhabiting stream substrates are known to be a sensitive and robust indicator of stream health and water quality [[15],[30]]. Consequently, changes in water quality can be tracked by monitoring changes in aquatic insect community composition. Because aquatic insects integrate stream water quality over time, they provide a more reliable measure of water quality than single-time-point chemical measurements. Aquatic insects are especially useful as biomonitors because (1) they are found in nearly all running-water habitats, (2) their large species diversity offers a wide range of responses to water quality change, (3) the taxonomy of most groups is well known and identification keys are available, (4) responses of many species to different types of pollution have been established, and (5) data analysis methods for aquatic insect communities are available [31]. Because of these advantages, biomonitoring using aquatic insects is routinely employed by federal, state, local, tribal, and private resource managers to track changes in river and stream health and to establish baseline criteria for water quality standards. Collection of aquatic insect samples for biomonitoring is inexpensive and requires relatively little technical training. However, the sorting and identification of insect specimens can be extremely time consuming and requires substantial technical expertise. As a result, aquatic insect identification is a major technical bottleneck for large-scale implementation of biomonitoring.

Larval stoneflies are especially important for biomonitoring because they are sensitive to reductions in water quality caused by thermal pollution, eutrophication, sedimentation, and chemical pollution. On a scale of organic pollution tolerance from 0 to 10, with 10 being the most tolerant, most stonefly taxa have a value of 0, 1, or 2 [13]. Because of their low tolerance to pollution, change in stonefly abundance or taxonomic composition is often the first indication of water quality degradation. Most biomonitoring programs identify stoneflies to the taxonomic resolution of family, although when expertise is available, genus-level (and occasionally species-level) identification is possible. Unfortunately, because of constraints on time, budgets, and availability of expertise, some biomonitoring programs fail to resolve stoneflies (as well as other taxa) below the level of Order. This results in a considerable loss of information and, potentially, in the failure to detect changes in water quality.

Although automated identification of all types of aquatic insects is a long-term goal of this research agenda, stoneflies are an ideal model group for the development of these methods for several reasons.

First, they encompass a wide range of identification challenges, from very easy (highly patterned, distinctive species) to very difficult (species complexes of nearly-indistinguishable taxa). Second, ontological changes in body size, patterning, and allometry present an identification challenge that must be overcome for any automated technique to be viable. Third, local and regional variability within species provides even further challenges for training identification algorithms. Finally, automated identification of stoneflies will be immediately useful to biomonitoring programs even before the technique is available for other aquatic insect orders.

### **2.1.1 Mechanical Manipulation & Imaging**

A fundamental problem in pattern recognition is to exploit variability between categories (e.g., taxa) while eliminating variability within categories. One important source of variability that can be eliminated is variation during image capture. To achieve consistent, repeatable image capture, we have designed and constructed a software-controlled mechanical stonefly larval transport and imaging apparatus that positions specimens under a microscope, rotates them (to obtain views from various angles), and photographs them with a digital camera. Using this apparatus, imaging rates of a few tens of specimens per hour can be achieved. A minimum of eight images (from different viewing angles) are taken of each specimen. The imaging apparatus has a series of mirrors so that each image acquires two simultaneous views of a specimen from approximately 90 degrees apart. Light diffusers reduce glare and eliminate hard shadows. In summary, the apparatus can quickly acquire several images of a specimen from various angles with consistent imaging conditions across specimens and species. Figure 2 shows the imaging apparatus, including the mirror setup used for acquiring two simultaneous images of each specimen. Figure 3 shows example images obtained using the imaging assembly.

## **2.2 Soil Mesofauna**

Agricultural and forest management is hampered by a lack of cost-effective methods for measuring insect populations and insect biodiversity. Such measurements can help society understand the impact of various forest and agricultural management practices on ecosystem health. Agricultural soils have been reduced in organic content worldwide, leading to a large portion of the carbon being lost to the atmosphere as CO<sub>2</sub> [[1],[38],[39]]. Numerous attempts to reverse this process have been tried in different cropping systems [8]. Tests for organic content in the soil are destructive and do not reveal anything about the biological diversity present. Population counts of soil mesofauna are recognized as one of the most sensitive, cheapest and least-destructive assays of soil biodiversity and nutrient cycling functions [41].

Biodiversity can be measured in many ways, in particular by assessing the population size, variety, and geographical distribution of key species. These species range in size from micro-organisms (bacteria and fungi) to megafauna (mammals and birds). For practical and theoretical reasons, arthropods (insects and their allies) are generally regarded as the best potential biodiversity indicators: their high species

richness allows for fine taxonomic resolution; their ease of capture results in low-cost assays; and their widespread distribution provides good generalizability and applicability [[6],[23],[28],[36]]. Historically, the practical limitations facing arthropod soil ecologists have been high densities 100 to 500,000/m<sup>2</sup> and high species diversity (a dozen in a 3 inch diameter soil core to hundreds in a meter squared). As a practical compromise, sample volume is always drastically reduced and the decreased number of replicate samples is often insufficient with respect to the environmental heterogeneity inherent in soil samples.

Many of the most significant characteristics of soil mesofaunal samples are changes which take place in repeated subsequent sampling under changing ecological conditions. The ability to scan over enough samples (or large enough samples) and to detect changes in the species present or their relative abundances are challenges that perhaps only automated identification systems are uniquely capable of solving. Coarse identification (to the family-level resolution) is often sufficient to assign ecological function and is attainable with a modest image database. Species-level identification within a functional group is not necessarily as important as being able to distinguish the presence of multiple species—i.e., correctly identifying 5 species of entomobryid springtails which necessitates an immense archive of pictures (an ultimate goal) is not as significant as recognizing that there are five different ‘morphospecies’ (a practical goal).

### **2.2.1 Mechanical Manipulation & Imaging**

Soil mesofauna specimens are routinely processed by Berlese funnel extraction. Berlese extraction involves slow drying of a soil sample, which causes the soil arthropods to crawl out of the soil and into a collecting vessel. The specimens are then separated from the associated debris by addition of an organic liquid followed by agitation, settling, and decanting [27]. Decanted samples are then placed in Petri plates and diluted sufficiently such that the probability of one specimen occluding another is low.

The mechanical manipulation of the soil mesofauna specimens is still under development (as indicated in Fig. 1); however, our current design is as follows. The Petri plate containing soil mesofauna specimens will be placed on a computer-controlled motorized x-y stage that allows the computer to scan the plate systematically to find each specimen. The specimens are so small (ranging from 50 to 2500 microns) that only a portion of each is in focus at any one time. This problem will be addressed by a montage process in which the focal plane of the microscope scans vertically through the field of view (via computer-control focus column) while a series of 10-20 images are taken. The images will then be combined to produce a synthetic image in which all parts of the specimens are simultaneously in focus. An additional benefit of this process is that it can produce an approximate 3D map of the height of each specimen, which may be useful for reconstructing the specimen’s 3D shape. After a specimen has been photographed, it can be removed mechanically from the Petri plate and placed in an appropriate receptacle to ensure that no specimen is counted twice and to provide a means of manual performance auditing.

Many of the soil arthropods are partially transparent or translucent. While specimen translucency may be an important cue for identification, it poses difficulties for automatically separating the specimens from

the image background. Image matting methods [[7],[19],[33],[40]] can be applied to determine the amount of transparency (on a per pixel basis) of a specimen if it is photographed with different background color patterns [34]. Hence, an LCD display panel will be placed between the Petri plate and the x-y stage to provide automatic control over the background color pattern.

## **2.3 Classification Framework**

Having motivated the classifications tasks and discussed the image acquisition methodology, we now address the problem of automatically identifying the family, genus, and species of each specimen. This proceeds in three steps: segmentation, coarse classification, and fine classification.

### **2.3.1 Segmentation**

The first step in classifying a specimen is to segment the image to separate a specimen from the background and, possibly, from other specimens in the frame. While segmentation is not required for identification, it does simplify many aspects of both learning and classifying objects. Without segmentation, the classification algorithm may confuse irrelevant background features (such as bubbles, dirt, etc.) with desired specimen features.

Automatic segmentation of a general class of images is a very difficult problem that remains unsolved. However, one advantage of the imaging mechanisms described above is that we can control the background, which greatly simplifies segmentation. Part of the image capture protocol for stoneflies is to acquire a background image (without a specimen) prior to positioning each new specimen under the microscope. This background image allows for a simple background subtraction process to provide a majority of the segmentation. For soil mesofauna, known background images are provided by the LCD display. In both cases, a Bayesian matting process [7] can be applied to improve segmentation and provide partial transparency for pixels near the specimen's boundary or for specimens that are translucent (as are some soil mesofauna species). Our current automatic segmentation algorithm performs most, but not all, of the work of segmentation. It is often still necessary to manually post-process the images to further refine the segmentation (such as remove bubbles, etc.).

### **2.3.2 Coarse Classification**

Our work to date has focused primarily on stoneflies. Since they share the same body plan and many have a similar general shape, there has been little need for performing coarse classification of stonefly larvae beyond manually identifying them as Plecoptera during specimen collection. However, we anticipate that the identification of soil mesofauna will greatly benefit from a coarse classification step that automatically groups the specimens with similar body plans, sizes, shapes and colors. Our plan is to extract, for each specimen, a set of shape cues such as eccentricity, compactness, and Fourier shape coefficients. We will also examine more detailed shape models that capture local curvatures of the specimen's silhouette, which may allow us to count protuberances such as legs, antennae, tails, and so

on [45]. Additionally, a specimen's color histogram information can also provide an indicator for separating specimens into groups. Using the shape and color information, each specimen will be classified into a general morphological group which will then provide the basis for fine classification within each such group.

### 2.3.3 Fine Classification

Identification to finer taxonomic levels will utilize both two-dimensional image information and three-dimensional reconstructions of the specimens. Two-dimensional methods operate directly on one or more images of the specimen, typically taken from preferred views. Our 2D approach is to extract various interest regions and construct invariant descriptors for each region. The descriptors are then clustered and a training algorithm learns feature-cluster associations. Details of 2D classification are provided in Section 4.

Three-dimensional methods apply various techniques to construct a 3D model of the specimen from 2D images. One technique is to fit a parameterized 3D model so that, when projected onto the 2D image plane, it produces an image that matches the image of the specimen. Another technique is to combine multiple 2D images to construct a 3D model of the specimen, which is then matched to 3D models. Our current approach is to create a 3D depth map of the specimens from multiple images with different focal planes—as mentioned previously in relation to creating images of the soil mesofauna specimens with all their body parts in focus. We believe that the resulting depth map will then allow us to extract 3D features that can be classified in a manner similar to the 2D features as discussed below.

## 3 Recent Methods for Object Recognition

Recent work in computer vision has led to the development of a new family of methods for object recognition. In this project, we are refining and extending these techniques so that they can be applied to recognize insects and soil arthropods. Hence, before describing our specific methods, we first review this family of modern object recognition methods. These methods are based on the following general approach:

**Step 1: Apply a 'region detector' to the input image.** Many recent object recognition methods classify objects by identifying and describing a collection of small regions or 'patches' in an image. These patch-based approaches work even when objects are partially occluded or when they are photographed in front of complex background settings. The first step in object recognition is to apply 'region detectors' to identify and extract a set of 'interesting regions' from the image. A *region detector* identifies a location (i.e., a pixel or region), a scale (i.e., a circle of specified radius or an ellipse with specified major and minor axes), and an orientation for each region. A good region detector should be reliable and informative. A detector is reliable if it is robust to changes in viewpoint, scale, illumination, and noise. That is, given two images of the same object taken with different viewpoints, illumination, and noise levels, the region detector should still detect the same regions. A detector is informative if the detected regions are

useful for discriminating among the different object classes. Some popular region detectors include the Harris [12], Harris-affine [26], maximal difference-of-Gaussian (DOG) [20], maximally stable extremal regions (MSER) [22], and entropy (or Kadir) [16] detectors. Region detectors are also known as interest operators because they find interesting (i.e., locally unique) points or regions within the image.

**Step 2: Represent each region by a fixed-length ‘descriptor’.** After detection, each interest region is represented by a *descriptor* (i.e., a real-valued vector of features) that succinctly and discriminatively characterizes the local image properties. By constructing each region’s descriptor relative to the local coordinate frame determined by the interest operator (i.e., the position, orientation, and scale), the descriptors are translation-, rotation-, and scale-invariant. Additionally, a descriptor should be insensitive to changes in illumination and, especially for object class recognition (as required in this project), non-rigid transformations. Recent descriptors include spin images [18], shape context [4], SIFT [21], and PCA-SIFT [17].

In this work we employ the SIFT (Scale Invariant Feature Transform) descriptor, which has been shown to perform better than other local descriptors [25]. The SIFT descriptor is invariant to scale, rotation, intensity and contrast changes, and, to a small degree, affine transformations. SIFT divides the region into a set of bins. For each bin, it computes a histogram of the intensity gradient orientation at each pixel. The result is a 128-dimensional real-valued vector. Once each detected region has been converted to a SIFT vector, the input image is discarded, and only the “bag” of SIFT vectors is retained for further analysis.

**Step 3: Apply a classifier to the bag of descriptor vectors.** Several classifiers have been developed that can analyze the bag of descriptor vectors and predict the class of the object. The simplest kind of classifier represents each object as a collection of ‘parts’. (We write ‘parts’ in quotes because these ‘parts’ do not necessarily correspond to real parts of the object; rather they correspond to interest regions in the object, which may or may not correspond to meaningful, physical parts of the object to be classified.) Each descriptor is classified according to which part it represents. If a sufficient number of matching parts is detected, the object is assigned to the corresponding class. More sophisticated classifiers (see below) compute a weighted sum of the detected parts. The most complex methods take into consideration the spatial relationships among the detected parts [[2],[10]], although subsequent work [[9],[29]] has obtained better results without including this information.

Representing an object by local salient regions has many advantages that have made region-based recognition very popular in recent years. Region-based representations cope better with images that have cluttered backgrounds and objects that are partially occluded. This is because classifiers can be trained to make a decision even if not all parts are detected. In cases where images need to be compared to one another, interest operators reduce the matching task from comparing hundreds of thousands (or millions) of pixels to comparing just a few hundred highly salient regions. Another major benefit of using local regions for object class recognition is that they provide a degree of object pose invariance. In particular,



affine-invariant detectors and descriptors allow for small out-of-plane object rotations (up to approximately 30°).

## 4 Classification Methods

As noted above, we are pursuing two strategies for insect classification: (1) recognition using features extracted from the two-dimensional image, and (2) recognition based on three-dimensional reconstruction of the specimen from multiple images. As our work on 3D reconstruction is still preliminary, this chapter reports only on the methods and results to date of our 2D approach.

Our 2D approach follows the general region-based methodology introduced above. This section presents the details of the approach. To date, we have focused only on discriminating between pairs of taxa (i.e., binary classifiers). Hence, we refer to one as the ‘positive’ class and the other as the ‘negative’ class. If we attain high accuracy on pair-wise discrimination, there are many machine learning methods for extending this to discriminate among tens or hundreds of taxa [3].

### 4.1 Region Detectors

We have experimented with many of the region detectors discussed previously, and we have chosen two: Harris-affine [26] and Kadir [16]. Our system constrains these detectors to consider only points that lie within the specimen, so that spurious detections in the background are not a problem. This is possible because the specimens have been segmented from the background.

### 4.2 Region Descriptors

For every interest region detected, we construct a 128-element SIFT vector that describes each region’s local neighborhood. The SIFT descriptor computes a 16×16 neighborhood centered on the detected region—normalized to scale, rotation, and (for Harris-affine) the affine parameters produced by the detector. This neighborhood is partitioned into 16 subregions of 4×4 pixels each. For each pixel within a subregion, SIFT adds the pixel’s gradient vector to a histogram of gradient directions by quantizing each orientation to one of 8 directions and weighting the contribution of each vector by its magnitude. Each gradient direction is further weighted by a Gaussian of scale  $s = n/2$  where  $n$  is the neighborhood size, and the values are distributed to neighboring bins using trilinear interpolation to reduce boundary effects that occur when pixels move across bin boundaries. The final descriptor is a 128-dimensional real vector—representing the 4×4 grid of 8-bin orientation histograms. Figure 4 shows a graphical representation of the SIFT descriptors created for three regions detected in two stonefly images. The SIFT vectors in Figures 4C and 4D are very similar, and these points are in corresponding positions on the two specimens. In contrast, the SIFT vector shown in Figure 4E is quite distinct, and it does not correspond to the other two points. This shows the ability of SIFT to capture distinctive characteristics of the detected regions.

The SIFT descriptor is invariant to scale, rotation, contrast and intensity changes, and small out-of-plane rotations. Scale invariance is achieved by describing the local neighborhood around each feature point at that feature's characteristic scale (as computed by the Harris or Kadir detectors). To achieve rotation invariance, the descriptor bins and gradient directions are defined relative to the dominant gradient orientation in the neighborhood. Invariance to intensity and contrast changes results from normalizing the 128-vector to unit magnitude. The 4x4 bin size (relative to the detected feature's characteristic scale) provides some invariance to minor affine and perspective transformations as well as some non-rigid distortion (such as typically occurs from interclass variability).

### 4.3 Defining Object Parts by Clustering Descriptor Vectors

The first two steps of detecting interesting regions and representing them by SIFT descriptor vectors are performed both during training of the classifier and during classification of new specimens. The third step—defining object parts—is only performed during training.

Parts are defined by performing a cluster analysis on the SIFT descriptor vectors extracted from the input images of the positive specimens. As does [9], we cluster the vectors by fitting a Gaussian Mixture Model (GMM). A GMM assumes that each SIFT vector is generated from one of  $K$  clusters according to a probability  $P(C_i)$ , where  $i = 1, \dots, K$  denotes the cluster number. The values of the 128-element descriptor vectors are assumed to have a Gaussian distribution within each cluster with mean vector  $\mu_i$  and diagonal covariance matrix  $\Sigma_i$ . According to this model, the probability assigned to any particular SIFT vector  $x$  can be computed as

$$p(x) = \sum_{i=1}^K p(x|C_i)P(C_i) \quad (\text{x.1})$$

where  $p(x | C_i) = \text{gauss}(x | \mu_i, \Sigma_i)$  is the multivariate Gaussian probability density function. The adjustable parameters in the model are  $\mu_i$ ,  $\Sigma_i$  and  $P(C_i)$  for  $i = 1, \dots, K$ . These parameters are fit by maximum likelihood (i.e., to maximize  $p(x)$  on the training data vectors  $x$ ) via the well-known Expectation-Maximization (EM) algorithm [24]. EM is initialized using the very efficient  $K$ -means algorithm, and it typically converges in 20 to 100 iterations.

To apply the GMM, we must choose a value for  $K$ . If  $K$  is too small, then the clusters found by EM will be very broad (with large variances in  $\Sigma_i$ ). This will cause them to fail to be distinctive or informative. On the other hand, if  $K$  is too large, then the clusters will overfit the training data and not generalize well. Values within the range  $50 \leq K \leq 100$  give the best results.

### 4.4 Converting Sets of SIFT Vectors into Standard Feature Vectors

Once the 'parts' have been defined, we then convert each set of SIFT vectors (extracted from one training image) into a single feature vector of length  $K$  as follows.

- 1) Initialize a histogram vector:  $hist[j] = 0$  for  $1 \leq j \leq K$ .

- 2) For each SIFT descriptor vector  $x$  extracted from the image:
  - a) Let  $i^* = \arg \max_i P(C_i|x) = \arg \max_i p(x|C_i) P(C_i)$
  - b)  $hist[i^*] = hist[i^*] + 1$ .
- 3) Normalize the histogram to unit magnitude.

Step 2a computes the ‘part’ cluster  $i^*$  that is most likely to have generated the observed SIFT vector  $x$  according to the GMM. Consequently, the  $i^{\text{th}}$  entry in the histogram vector is proportional to the number of SIFT vectors that “belong” to cluster  $i$ .

This conversion gives us training data in a format suitable for analysis by standard supervised learning algorithms. Let us denote the normalized histogram for the  $j^{\text{th}}$  input image by  $h_j$  and denote the corresponding class label (taxon) by  $y_j$ . Our training data thus consist of pairs  $(h_j, y_j)$  for  $j = 1, \dots, N$ , where  $N$  is the total number of training images.

#### 4.5 Training the Classifier

After the SIFT vector sets have been converted to standard feature vectors, they are used to train a classifier. In this chapter, we report results of a ‘bagged’ decision tree classifier. A decision tree classifier has the form of a nested set of **if-then-else** statements, where each statement has the form

```

if  $h[i] > \theta_n$ 
  then statement
  else statement.

```

and *statement* can either be a nested **if-then-else** or else a predicted class label. A new specimen is assigned a predicted class by executing this tree of **if-then-elses** until a predicted class label is reached.

Decision tree classifiers are learned top-down by first selecting a feature  $i$  and a threshold  $\theta_i$  for the outer-most **if-then-else** and then splitting the training data according to the results of this test. If all of the examples that reach a *statement* belong to a single class, then the recursion halts and that class is predicted. Similarly, if a new **if-then-else** would result in sending two or fewer examples down either the **then** or the **else** branches, then the recursion halts, and the class belonging to a majority of the data points is assigned at that *statement*. We have employed the J48 decision tree learning algorithm, which is part of the WEKA machine learning system [44].

A single decision tree is typically not a very good classifier. However, very high performance can often be obtained by constructing an ensemble of decision trees through a method known as ‘bagging’ [5]. In bagging, the decision tree learning algorithm is applied to  $L$  different training sets. Each training set is constructed by drawing  $N$  examples uniformly, with replacement, from the original training data set. Such a training set is known as a bootstrap replicate. Each bootstrap replicate may contain multiple copies of some of the original training examples, and it may be missing other components of the original examples. On average, a bootstrap replicate contains approximately 62 percent of the original training

examples (but with enough copies so that there are still  $N$  total examples). In the results reported below bagging has been applied to construct 15 decision trees.

To classify a new example, the predictions of these 15 decision trees are computed and the class (taxon) with the largest number of predictions is chosen as the overall bagged prediction.

## 5 Results and Discussion

We report the results of three experiments on stonefly classification. In each experiment, we train our method to discriminate between two species. The three experiments present progressively increasing levels of difficulty:

1. The somewhat easy task of discriminating between the very distinctive *Calineuria californica* and *Yoraperla sp.*
2. A moderately difficult task of discriminating between the *Hesperoperla pacifica* from *Doroneuria baumanni*.
3. A hard task of discriminating between the very similar *Calineuria californica* and *Doroneuria baumanni*.

These experiments are performed as follows: the image dataset is randomly divided into three completely disjoint sets of equal size. To avoid any kind of influence in the test results, different images of the same insect instance are placed in the same set. The first set is used as the ‘clustering set’ to create the GMM clusters for each object class (as detailed in Section 4.3 above); the second set is used to train the 15 decision trees that comprise the final classifier (described in Sections 4.4 and 4.5); while the third is used to measure the classification accuracy of the classifier. We use separate clustering and training sets to reduce overfitting of the classifier to the training data.

Table I presents the resulting classification accuracy rates for our three experiments (along with 95% confidence intervals). For experiment 1 (*Calineuria* vs. *Yoraperla*), the method achieved 94% accuracy; for experiment 2 (*Hesperoperla* vs. *Doroneuria*), the accuracy was 90%, and for the very difficult experiment 3 (*Calineuria* vs. *Doroneuria*), the method attained only 73% correct classifications. These results represent the best matching rates achieved thus far in differentiating between pairs of insect species. However, because some of our design decisions (number of clusters, choice of species for clustering, number of decision trees to combine) were made to optimize these numbers, they are probably optimistically high.

Because these are binary classification experiments, either species’ clustering set can be used to learn the GMM part clusters. In our experiments, the GMM clusters of both species were tried and the one that rendered the best results is presented. *Calineuria*’s part clusters were employed in the first experiment, *Hesperoperla*’s in the second experiment, and *Doroneuria*’s in the third. The choice of clustering set affects the accuracy, and there are two possible reasons for this: (1) some species have more clustering examples in the data set than others and (2) some species exhibit more distinctive visual cues that help with recognition. In the first experiment, the dataset contains many more images of

*Calineuria* than of *Yoroperla*, this difference probably accounts for the GMM clustering algorithm producing better *Calineuria* part clusters which in turn helped increase the classification rate. By contrast, the data set in experiment three contains similar numbers of *Calineuria* and *Doroneuria* images, and both clustering sets give similar results. In the second experiment, the high classification accuracy is surprising. It is possible that the *Hesperoperla* clustering set has certain visual cues that produce very distinctive GMM clusters.

Tables II, III, and IV present the confusion matrices of experiments 1, 2, and 3, respectively. Table II shows that all of the 16 misclassification errors involved misclassifying a *Calineuria* as a *Yoraperla*, which is quite surprising, because normally, misclassification favors the class with the larger number of training examples (in this case, *Calineuria*). Table III shows a similar pattern: most of the errors misclassify *Hesperoperla* as *Doroneuria*. Finally, in Table IV, we see a more balanced ratio of errors. The poor results in experiment 3 accord with our own informal experience: both *Calineuria* and *Doroneuria* are from the Perlidae family and look very similar—so much so that the non-entomologists on our team cannot generally distinguish them using only the collected image data. While *Hesperoperla* is also in the Perlidae family, it is more visually distinct from *Calineuria* and *Doroneuria*.

## 6 Concluding Remarks and Future Work

This chapter describes our ongoing work on developing robust pattern recognition methods for automatically classifying insects. Although this project focuses on identifying stonefly larvae for stream water quality monitoring and soil mesofauna for soil biodiversity assessment, the goal is to create general-purpose techniques that can be applied to many automated systematics tasks by simply retraining the system for each application. In addition, we have designed and developed (or are developing) computer-controlled mechanical systems to facilitate automated imaging of specimens. Our insect classification system uses recent advances in object-class recognition based on interest operators and bagged local region descriptors combined with boosted decision trees.

While we are pleased with the classification accuracy achieved thus far on stonefly larvae, there is still significant room for improvement. First and foremost, we continue to add and refine components that, as indicated in Fig. 1, are still under development. These components include the control and imaging software for the soil mesofauna, fully automate segmentation by incorporating Bayesian matting (or a similar method), coarse classification, and the extraction and inclusion of 3D features from the depth map constructed during the focal-montage process. In addition to these necessary components, we continue to make improvements to the existing mechanical specimen manipulation and imaging apparatus with the ultimate goal of mechanically delivering each identified specimen to a separate bin based on its classification. We are also developing a new watershed-based region detector [46] that appears to be better suited for insect classification and we continue to explore additional region and feature descriptors to supplement the SIFT feature vectors. Furthermore, we are experimenting with several additional classifiers including those that work directly on the bags of descriptor vectors, instead of converting them

to standard feature vectors. We are also developing a multi-class system that will classify a specimen into one of many taxonomic categories rather than our current binary classifiers that distinguish only between two species at a time.

Our current effort has focused on classifying each specimen to the species level (though in the experiments presented here, genus-level classification implies species-level identification as well) . However, in cases where the classifier is uncertain about the species, it is often useful (both for stream monitoring and for soil biodiversity studies) to make a coarser classification at the level of genus or family. For example, while it is difficult to achieve high confidence in automatically distinguishing between the very similar *Calineuria californica* and *Doroneuria baumanni* species of stonefly larvae, high confidence classification to the family level is much more feasible and still very valuable for stream health assessment. Hence, we plan to explore methods for trading off the benefit of fine classification against the risk of making an error in order to automatically choose the best level of the taxonomic hierarchy for classifying each specimen.

We would also like to explore additional systematics applications to test the robustness of our system in new classification domains, such as recognition of plant species. Our long-term goal is to develop commercial products to make this emerging technology available to the environmental monitoring and research communities.

## 7 Acknowledgments

The authors gratefully acknowledge the support of the US National Science Foundation under grant number IIS-0326052. The views expressed are those of the authors and do not represent the views of the US Government or the National Science Foundation.

## 8 References

- [1] Adl, S. M., *The Ecology of Soil Decomposition*, CABI Publishing, 2003.
- [2] Agarwal, S. and Roth, D., Learning a sparse representation for object detection. *Proc. European Conference Computer Vision*, pp. 113-128, 2002.
- [3] Allwein, E. L., Schapire, R. E., and Singer, Y., Reducing multiclass to binary: a unifying approach for margin classifiers, *J. Machine Learning Research*, **1**:113-141, 2000.
- [4] Belongie, S., Malik, J., and Puzicha, J., Shape matching and object recognition using shape contexts, *IEEE Trans. Pattern Analysis Machine Intelligence*, **24**(4):509-522, 2002.
- [5] Breiman, L., Bagging predictors, *Machine Learning*, **24**(2):123-140, 1996.
- [6] Brown, K. S., Diversity, disturbance and sustainable use of neotropical forests: insects as indicators for conservation monitoring, *J. Insect Conservation*, **1**(1):25-42, 1997.

- [7] Chuang, Y.-Y. et al., A Bayesian approach to digital matting, in *Proc. Computer Vision and Pattern Recognition*, Vol. II, pp. 264-271, 2001.
- [8] Coleman, D. C. and Hendrix, P. F., *Invertebrates as Webmasters in Ecosystems*, CABI Publishing, 2000.
- [9] Dorkó, G. and Schmid, C., Object class recognition using discriminative local features, *Technical Report RR-5497*, INRIA – Rhone-Alpes, 2005.
- [10] Fergus, R., Perona, P., and Zisserman, A., Object class recognition by unsupervised scale-invariant learning, in *Proc. Computer Vision and Pattern Recognition*, Vol. II, pp. 264-271, 2003.
- [11] Gaston, K. J. and O'Neill, M. A., Automated species identification: why not?, *Philosophical Trans. Royal Soc. B: Biological Sciences*, **359**(1444):655-667, 2004.
- [12] Harris, C. and Stephens, M., A combined corner and edge detector, in *Proc. of the Fourth Alvey Vision Conf.*, pp. 147-151, 1988.
- [13] Hilsenhoff, W. L., Rapid field assessment of organic pollution with a family level biotic index, *J. North American Benthological Soc.*, **7**(1):65-68, 1988.
- [14] Jalba, A. C. et al., Automatic diatom identification using contour analysis by morphological curvature scale spaces, *Machine Vision and Applications*, **16**(4):217-228, 2005.
- [15] Johnson, R. K., Wiederholm, T., and Rosenberg, D. M., Freshwater biomonitoring using individual organisms, populations, and species assemblages of benthic macroinvertebrates, in *Freshwater Biomonitoring and Benthic Macroinvertebrates*, D. M. Rosenberg and V. H. Resh (eds.), pp. 40-158, Chapman & Hall, 1993.
- [16] Kadir, T. and Brady, M., Saliency, scale and image description, *Int'l J. of Computer Vision*, **24**(2):83-105, 1996.
- [17] Ke, Y. and Sukthankar, R., PCA-SIFT: A more distinctive representation for local image descriptors, in *Proc. Computer Vision and Pattern Recognition*, Vol. I, pp. 511-517, June 2004.
- [18] Lazebnik, S., Schmid, C., and Ponce, J., Sparse texture representation using affine-invariant neighborhoods, In *Proc. Computer Vision and Pattern Recognition*, Vol. II, pp. 319-324, 2003.
- [19] Levin, A., Lischinski, D., and Weiss, Y., A closed form solution to natural image matting, in *Proc. Computer Vision and Pattern Recognition*, 2006.
- [20] Lowe, D. G., Object recognition from local scale-invariant features, in *Proc. International Conf. on Computer Vision*, pp. 1150-1157, 1999.
- [21] Lowe, D. G., Distinctive image features from scale-invariant keypoints, *Int. Journal Computer Vision*, **60**(2):91-110, 2004.

- [22] Matas, J. et al., Robust wide-baseline stereo from maximally stable extremal regions, in *Proc. British Machine Vision Conf.*, pp. 384–393, 2002.
- [23] McGeoch, M. A., The selection, testing and application of terrestrial insects as bioindicators, *Biological Reviews of the Cambridge Philosophical Soc.*, **73**:181-201, 1998.
- [24] McLachlan, G. and Krishnan, T. *The EM algorithm and extensions*, New York: Wiley, 1997.
- [25] Mikolajczyk, K. and Schmid, C., A performance evaluation of local descriptors, in *Proc. Computer Vision and Pattern Recognition*, Vol. II, pp. 257-264, 2003.
- [26] Mikolajczyk, K. and Schmid, C. Scale and affine invariant interest point detectors, *Int'l J. Computer Vision*, **60**(1):63-83, 2004.
- [27] Moldenke, A. R., Arthropods, in *Methods of Soil Analysis, Part 2, Microbiological and Biochemical Properties*, SSSA Book Series, No. 5, pp. 517-541, 1994.
- [28] Niemelä, J., et al., Carabid beetle assemblages (Coleoptera, Carabidae) across urban-rural gradients: an international comparison, *Landscape Ecology*, **17**(5):387-401, 2002.
- [29] Opelt, A. et al, Weak hypotheses and boosting for generic object detection and recognition, in *Proc. European Conf. Computer Vision*, Vol. II, pp. 71–84, 2004.
- [30] Resh, V. H. and Jackson, J. K., Rapid assessment approaches to biomonitoring using benthic macroinvertebrates, in *Freshwater Biomonitoring and Benthic Macroinvertebrates*, D. M. Rosenberg and V. H. Resh (eds.), pp. 195-233, Chapman & Hall, 1993.
- [31] Resh, V. H., Myers, M. J., and Hannaford, M. J., Macroinvertebrates as biotic indicators of environmental quality, in *Methods in Stream Ecology*, F. R. Hauer and G. A. Lamberti (eds.), pp. 647-667, Academic Press, 1996.
- [32] Roth, V. et al., Integrating feature-based and pixel-based classification for the automated identification of solitary bees, in *Jahrestagung der Deutschen Gesellschaft für Mustererkennung (Annual Convention of the German Society for Pattern Recognition)*, pp. 120-129, 1999.
- [33] Ruzon, M. and Tomasi, C., Alpha estimation in natural images, in *Proc. Computer Vision and Pattern Recognition*, Vol. I, pp. 18-25, 2001.
- [34] Smith, A. R. and Blinn, J. F., Blue screen matting, in *Proc. Computer Graphics*, pp. 259-268, 1996.
- [35] Steinhage, V. et al., ABIS: automated identification of bee species, in *BIOLOG Workshop*, 2001.
- [36] Szaro, R. C. and Johnston, D. W., *Biodiversity in Managed Landscapes*. Oxford Univ. Press, 1996.
- [37] Turk, M. and Pentland, A., Eigenfaces for recognition, *Journal of Cognitive Neuroscience*, **3**(1):71-86, 1991.



- [38] van der Putten, W.H., et al., The sustainable delivery of goods and services provided by soil biota, in *Sustaining Biodiversity and Ecosystem Services in Soil and Sediments*, D.H. Wall (ed.), pp. 15-44, Island Press, 2004.
- [39] Wall, D. H., *Sustaining Biodiversity and Ecosystem Services in Soils and Sediments*, Island Press, 2004.
- [40] Wang, J. and Cohen, M., An iterative optimization approach for unified image segmentation and matting, in *Proc. Intl. Conf. on Computer Vision*, Vol. II, pp. 936-943, 2005.
- [41] Wardle, D. A., *Communities and Ecosystems: Linking the Aboveground and Belowground Components*, Princeton University Press, 2002.
- [42] Watson, A. T., O'Neill, M. A., and Kitching, I. J., Automated identification of live moths (Macrolepidoptera) using digital automated identification system (DAISY), *Systematics and Biodiversity*, **1**(3):287-300, 2003.
- [43] Webb, G., Multiboosting: a technique for combining boosting and wagging, *Machine Learning*, **40**(2):159-196, 2000.
- [44] Witten, I. H. and Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*, Morgan Kaufmann, 2005.
- [45] Wu, P. and Dietterich, T. G., Improving SVM accuracy by training on auxiliary data sources. *Int'l Conf. Machine Learning*, pp. 871-878, 2004.
- [46] Zhang, W., Deng, H., Dietterich, T. G., and Mortensen, E. N., A hierarchical object recognition system based on multi-scale principal curvature regions, in *Proc. of Intl. Conf. Pattern Recognition*, 2006.

**TABLE I:** CLASSIFICATION RESULTS USING DECISION TREES WITH BAGGING

Experiment	Classification Accuracy (%)
1: <i>Calineuria</i> vs. <i>Yoraperla</i>	94.40±2.66
2: <i>Hesperperla</i> vs. <i>Doroneuria</i>	90.47±3.36
3: <i>Calineuria</i> vs. <i>Doroneuria</i>	73.33±6.88

**TABLE II:** CONFUSION MATRIX — *CALINEURIA* VS. *YORAPERLA*

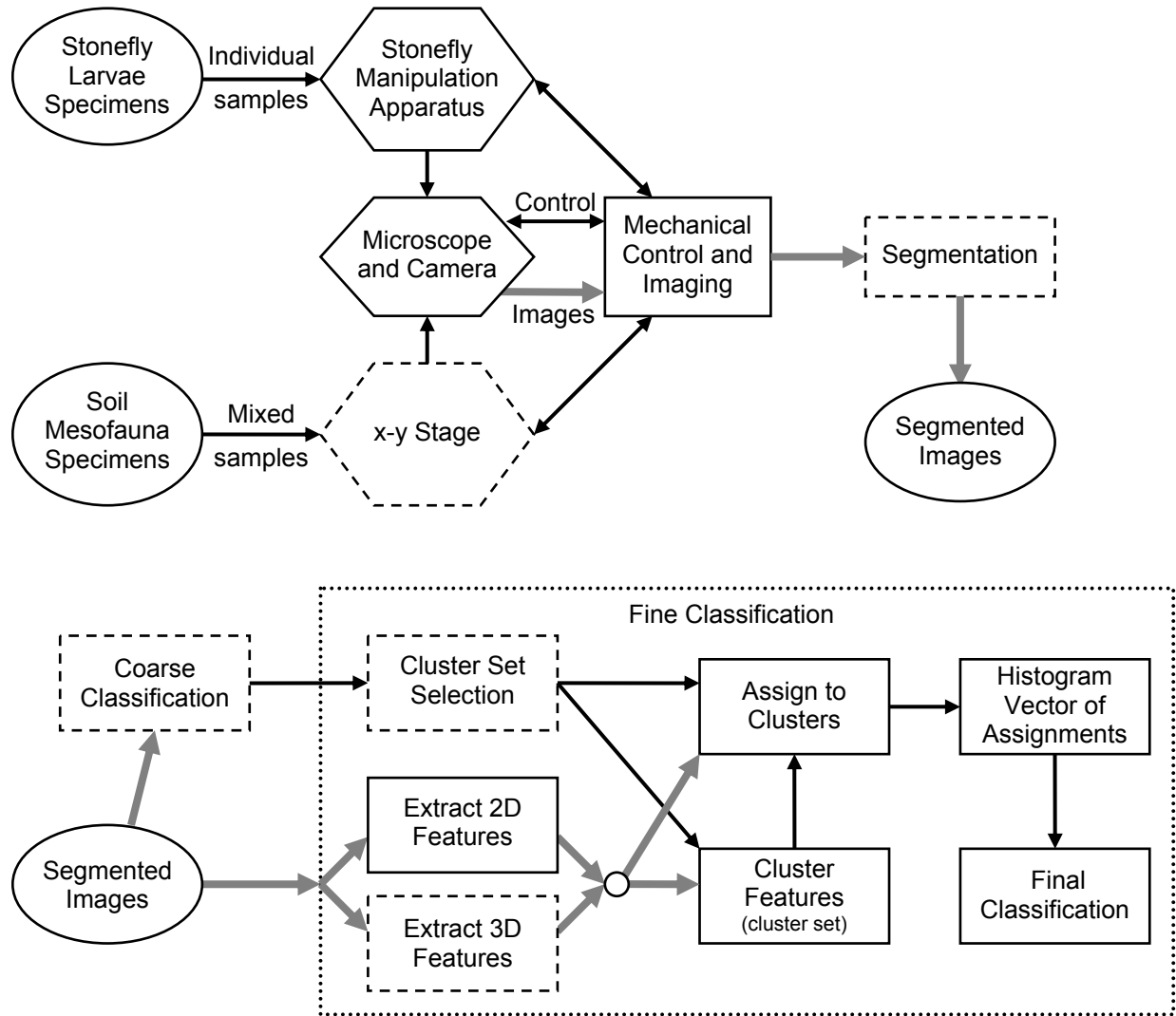
Taxa	Classified as: <i>Calineuria</i>	Classified as: <i>Yoraperla</i>
<i>Calineuria</i>	171	16
<i>Yoraperla</i>	0	99

**TABLE III:** CONFUSION MATRIX — *HESPEROPERLA* VS. *DORONEURIA*

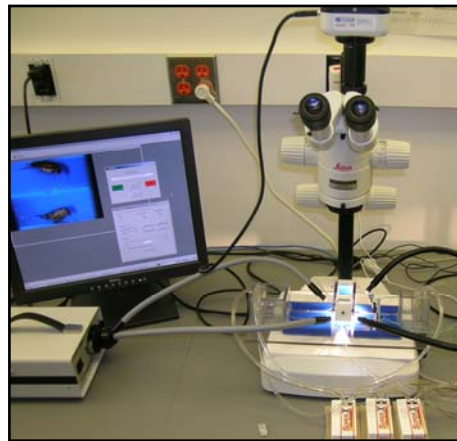
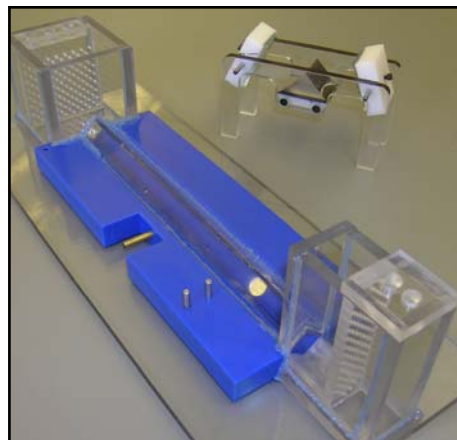
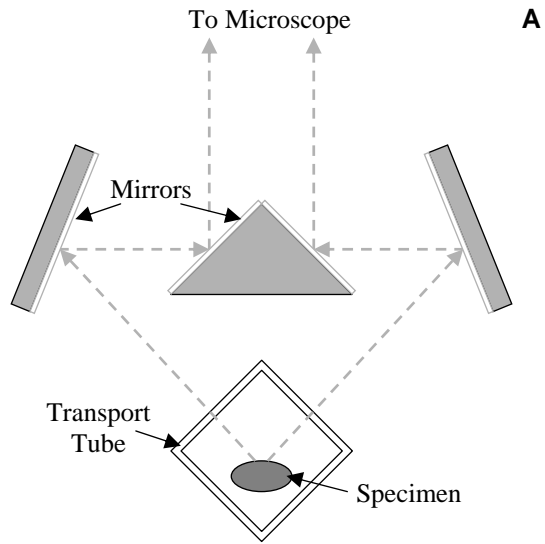
Taxa	Classified as: <i>Hesperoperla</i>	Classified as: <i>Doroneuria</i>
<i>Hesperoperla</i>	171	24
<i>Doroneuria</i>	4	95

**TABLE IV:** CONFUSION MATRIX — *CALINEURIA* VS. *DORONEURIA*

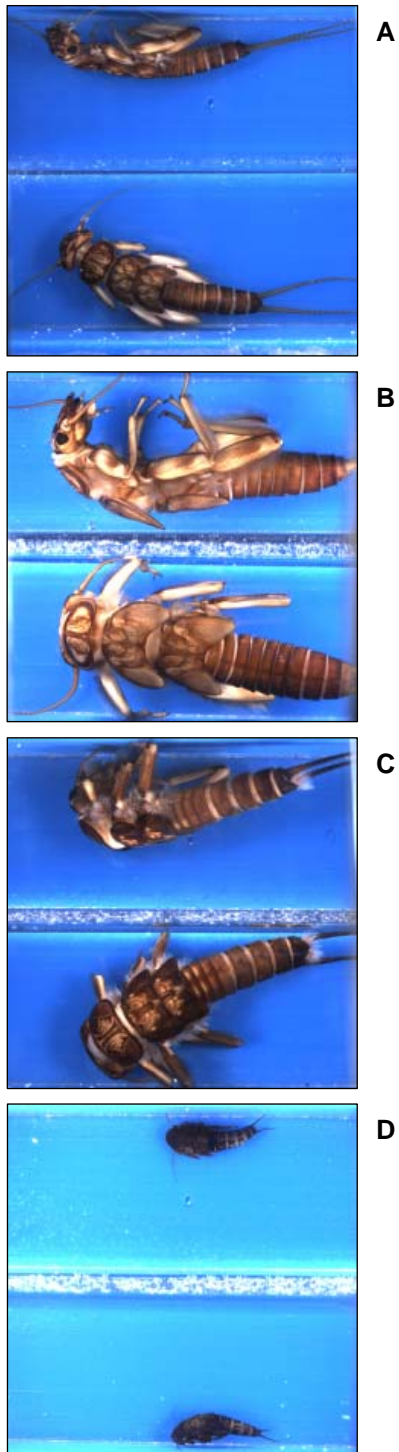
Taxa	Classified as: <i>Calineuria</i>	Classified as: <i>Doroneuria</i>
<i>Calineuria</i>	126	66
<i>Doroneuria</i>	30	138



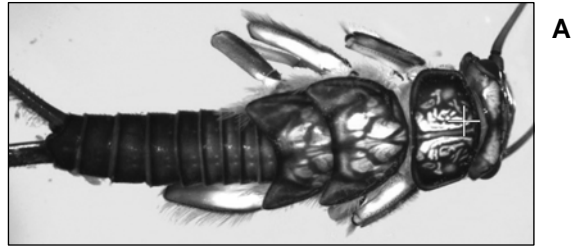
**Figure 1:** Diagram of insect classification system. Components with dashed lines are still under development.



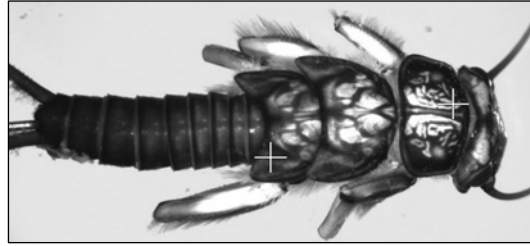
**Figure 2:** Transportation and imaging apparatus for stonefly larvae. A. Diagram of mirror system for obtaining two simultaneous views of a specimen (from approx. 90° apart) in a single image. B. Image of prototype mirror and transportation apparatus. C. Image of entire stonefly transportation and imaging setup (with microscope and attached digital camera, light boxes, and computer controlled pumps for transporting and rotating the specimen).



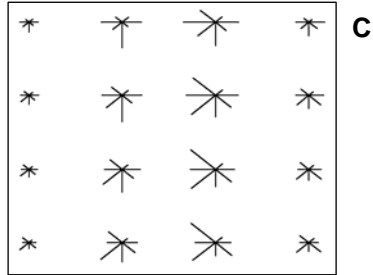
**Figure 3:** Example images of stonefly specimens taken with our imaging apparatus. A. *Calinueria californica*. B. *Doroneuria baumanni*. C. *Hesperoperla pacifica*. D. *Yoraperla* sp.



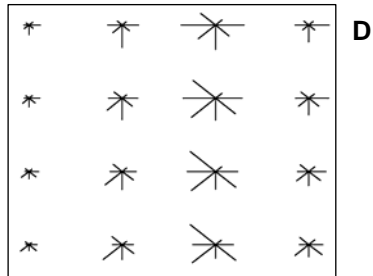
A



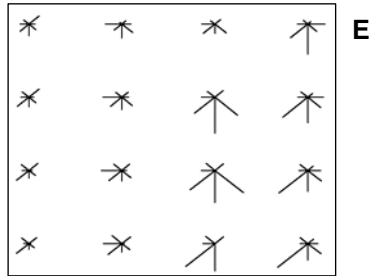
B



C



D



E

**Figure 4:** Example SIFT descriptors for three points in two images. The SIFT histograms for (c) the detected region in image (a), the matching region in image (b) and (e) a random region in image (b).