ARTICLE IN PRESS

Pattern Recognition Letters xxx (2013) xxx–xxx

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

# Structured analysis of the ISI Atomic Pair Actions dataset using workflows

Ricky J. Sethi *, Hyunjoon Jo, Yolanda Gil

*Information Sciences Institute, University of Southern California, 4676 Admiralty Way, Suite 1001, Marina del Rey, CA 90292, United States*

ARTICLE INFO

*Article history:*
Available online xxxx

*Keywords:*
Atomic Pair Action
Workflows
Human activity recognition

ABSTRACT

Understanding pair-wise activities is an essential step towards studying complex group and crowd behaviors in video. However, such research is often hampered by a lack of datasets that concentrate specifically on Atomic Pair Actions; [Here, we distinguish between the atomic motion of individual objects and the atomic motion of pairs of objects. The term action in Atomic Pair Action means an atomic interaction movement of two objects in video; a pair activity, then, is composed of multiple actions by a pair or multiple pairs of interacting objects (Ahad, 2011; Turaga et al., 2008). Please see Section 1 for details.] in addition, the general dearth in computer vision of a standardized, structured approach for reproducing and analyzing the efficacy of different models limits the ability to compare different approaches. In this paper, we introduce the ISI Atomic Pair Actions dataset, a set of 90 videos that concentrate on the Atomic Pair Actions of objects in video, namely converging, diverging, and moving in parallel. We further incorporate a structured, end-to-end analysis methodology, based on workflows, to easily and automatically allow for standardized testing of state-of-the-art models, as well as inter-operability of varied codebases and incorporation of novel models. We demonstrate the efficacy of our structured framework by testing several models on the new dataset. In addition, we make the full dataset (the videos, along with their associated tracks and ground truth, and the exported workflows) publicly available to the research community for free use and extension at <http://research.sethi.org/ricky/datasets/>.



## 1. Introduction

Group and crowd analysis is a growing area of study in computer vision. However, in order to examine group and crowd activities, it is necessary to first understand pair activities (Zhou et al., 2008, 2011; Gaur et al., 2009; Sethi and Roy-Chowdhury, 2010a). Pair activities, in turn, can be thought of as being composed of Atomic Pair Actions. Following the formulation of Turaga et al. (2008) and Ahad (2011), we use the term **Atomic Pair Action** to refer to simple, uniform motion patterns involving two, possibly interacting, objects in video, typically lasting for short durations of time. In contrast, the term **Pair Activity** refers to a complex sequence of Atomic Pair Actions performed by a pair, or multiple pairs, of interacting objects, typically characterized by much longer temporal durations.

Although, as Turaga et al. (2008) notes, there is no hard boundary between an action and an activity, most researchers (Wang et al., 2010; Zhang et al., 2006; Sethi and Roy-Chowdhury, 2010a,b; Oliver et al., 2000, 1999) refer to three basic motions as being fundamental pair motions. Thus, we also define Atomic Pair Actions to be two objects exhibiting motion towards each other (*converging*), moving away from each other (*diverging*), or moving in the same direction (*parallel*), as seen in Fig. 1. The motion of the individuals in these atomic pair motions might or might not be synchronized or aligned; e.g., in the case of the parallel Atomic Pair Motion (Fig. 1c), one object might be in front of the other object resulting in motion characterized as *following*, a derivative of the parallel Atomic Pair Action (Zhou et al., 2008, 2011; Ni et al., 2009).

Since pair-wise activities are composed of multiple Atomic Pair Actions, we need to examine Atomic Pair Actions as a precursor to examining more complex pair and, ultimately, group and crowd activities. Examples of Atomic Pair Actions include two people walking together, a person heading towards a car, etc. Pair-wise or Group Activities, on the other hand, are seen in scenarios like a soccer team scoring a goal, people fighting, students going to a lecture, etc. (Ni et al., 2009; Oliver et al., 2000; Gaur et al., 2009).

### 1.1. Problems and motivation

However, there is a lack of datasets for evaluating pair activities or Atomic Pair Actions. Most researchers rely on either creating a custom dataset or using parts of pre-existing datasets for the subset of activities they want to examine (Zhou et al., 2011, 2008; Ni et al., 2009; Gaur et al., 2009; Sethi and Roy-Chowdhury, 2010a; Nascimento et al., 2005) or utilizing simulations and synthetic

* Corresponding author. Tel.: +1 310 448 8678; fax: +1 310 823 6714.
*E-mail addresses:* rickys@sethi.org (R.J. Sethi), hyunjoon@usc.edu (H. Jo), gil@isi.edu (Y. Gil).

0167-8655/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.12.015

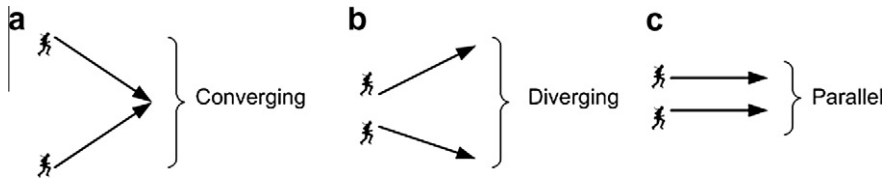



**Fig. 1.** Atomic Pair Actions.

agents (Oliver et al., 1999, 2000; Nascimento et al., 2005; Sethi and Roy-Chowdhury, 2010a); as noted by these researchers, having an appropriate dataset is a significant roadblock to thorough evaluation and development of their methods.

In addition, existing datasets confuse simple activities with atomic actions, as simple (or complex) pair activities can often be de-composed into their atomic components. For example, we note the *following* pair activity is a derivative of the *parallel* Atomic Pair Action and the *meeting* pair activity is composed of the *converging* + *parallel* + *diverging* Atomic Pair Actions (Zhou et al., 2011, 2008; Ni et al., 2009); similarly, activities like *chasing* (*converging* + *diverging*), *together* (*parallel*), etc., can all be characterized by their component Atomic Pair Actions.

Thus, most of the existing datasets are limited as they confuse Atomic Pair Actions and Pair Activities, use ad hoc components of other datasets, or create a dataset or simulations to demonstrate their specific method only. The ISI Atomic Pair Actions dataset aims to fill this gap and provide a dataset that concentrates solely on Atomic Pair Actions but is not targeted toward any specific method; in fact, another contribution of this article is a comparative analysis of many different methods on this novel dataset.

The third contribution of this article is in providing a way to easily reproduce, extend, and compare methods to analyze computer vision datasets. Such a framework is needed not only because there is a dearth of datasets that concentrate on Atomic Pair Actions but because, in general, there is no standardized, structured approach for analyzing the efficacy of the different models that examine such activities in computer vision. In fact, due to custom code implementations and the lack of a standardized structure to integrate heterogenous codebases, the ability to reproduce results of computational algorithms is a significant issue in science (Vandewalle et al., 2009).

By using the power of workflows, we provide just such a framework to standardize extensions and comparisons of any computer vision algorithm. Using a workflow-based framework allows for quick deployment and comparison of various approaches and algorithms. To the best of our knowledge, this is the first structured framework for easy integration, replication, and comparison of different computer vision methods and approaches.

### 1.2. Contributions

Our main contributions in this article are as follows:

- We provide a comprehensive new dataset for Atomic Pair Actions.
- We introduce a workflows-based methodology for evaluating state-of-the-art models for pair-wise analysis of Atomic Pair Actions, thus providing a structured, systematic framework for comparison of general computer vision models.
- Finally, we provide a comparative analysis of different approaches to analyze Atomic Pair Actions.

Thus, although our main contribution is a new dataset, we also present a structured and extensible analysis framework based on workflows and provide a comparative assessment of state-of-the-art pair analysis methods on the new dataset.

### 1.3. Overview of article

The rest of this article is organized as follows: in Section 2, we review the work related to pairwise activities and give some essential background on workflows. Next, in Section 3, we describe our new dataset and show some examples. In Section 4, we describe our implementation details, including the models used, the workflows generated, and parameter optimizations utilized. Finally, in Section 5, we give both the results for the individual models and the cumulative summary statistics for all the models utilized on our new dataset.

## 2. Related work

### 2.1. Activity recognition overview

Activity recognition is the task of interpretation of the activities of objects in video over a period of time. The goal of an activity recognition system is to extract information about the movements of objects and/or their surroundings from the video data in order to draw conclusions about the events and context in video in an automated manner. In a simple scenario where the video is segmented to contain only one execution of a human activity, e.g., an Atomic Pair Action, the objective of the system then becomes to correctly classify the activity into its category, whereas in a more complex scenario of a long video sequence containing multiple activities, it may also involve detection of the starting and ending points of all occurring activities in the video (Aggarwal and Ryoo, 2012; Aggarwal and Cai, 1999).

Human activities can be categorized into four classes: kinesics (movement of people or objects), proxemics (the proximity of people with each other or objects), haptics (people–object contact), and chronemics (change with time). Most work in activity recognition (Turaga et al., 2008; Aggarwal and Ryoo, 2012; Nascimento et al., 2005; Nayak et al., 2011; Sethi et al., 2011) deals with actions of humans that, even for multiple activities (Duchenne et al., 2009; Ryoo and Aggarwal, 2009), fall in the domain of kinesics and haptics. Atomic Pair Actions, on the other hand, rest mainly in the domain of proxemics and chronemics since they model the spatio-temporal relationships between multiple, interacting objects. We thus consider Atomic Pair Actions between two objects in the domain of chronemics and proxemics, e.g. within the context of surveillance videos.

### 2.2. Existing approaches for analyzing pair-wise motion

Most approaches to analyzing pair-wise motion (Wang et al., 2010; Zhang et al., 2006; Sethi and Roy-Chowdhury, 2010a,b; Oliver et al., 1999, 2000; Ni et al., 2009; Zhou et al., 2011) also rely upon chronemics and proxemics. Researchers such as Oliver et al. (1999, 2000) have utilized relative distances within a coupled HMM to look at pair activities, examining such motion as two people coming together, one person following another, etc. However, they only interpret pedestrian actions, rely on

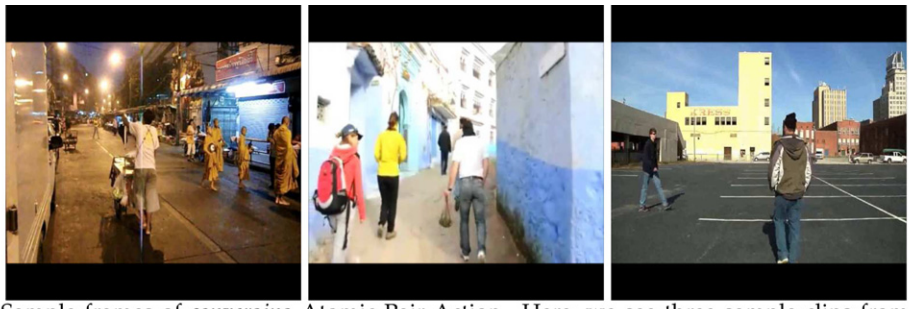
Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015

exhaustive training, and create prior models of human behavior by using synthetic agents that encapsulate their assumptions. Gaur et al. (2009) also examines pair-wise motions such as two objects following each other, two people meeting, etc., by conducting a heuristic examination of simple activities in a single domain using relative distances and also relying upon a classifier (a two-class nearest neighbor classification with extensive parameterizations and thresholds). (Zhou et al., 2008, 2011; Ni et al., 2009) look at activities such as walking together, chasing, meeting, etc.; they both use classifiers, as well, but take a unique approach of using a digital filter representation for causality analysis that relies on computing the Granger Causality Test. Wang et al. (2010) examines activities such as people coming together, people diverging, etc., and also relies upon a classifier by using a one-vs-all svm as well as a rule-based classifier. Finally, Sethi and Roy-Chowdhury (2010a,b) also examine pair actions such as objects converging, objects diverging, etc., but does not rely upon a classifier, instead using a physics-based approach to fit an exponential model, based on the second-order differential equation of motion for a damped oscillator.

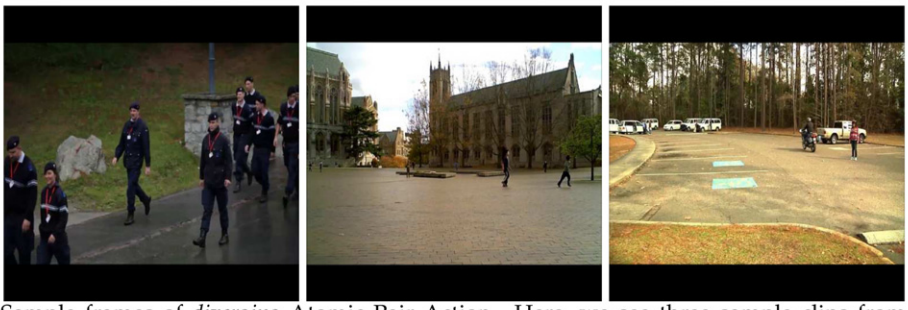
### 2.3. Background on workflows and Wings

Most of these varied approaches rely upon the same classifiers yet have widely divergent implementations making replication and reproducibility a nightmare. Using a semantic workflow system like Wings to assist with the design of such computational experiments allows for creating structured, systematic experiments that can be automated, thus allowing anyone to reproduce results, regardless of how parameters and thresholds are optimized for specific datasets. In addition, the Wings workflow system has an open modular design and can be easily integrated with other existing workflow systems and execution frameworks to extend them with semantic reasoning capabilities.

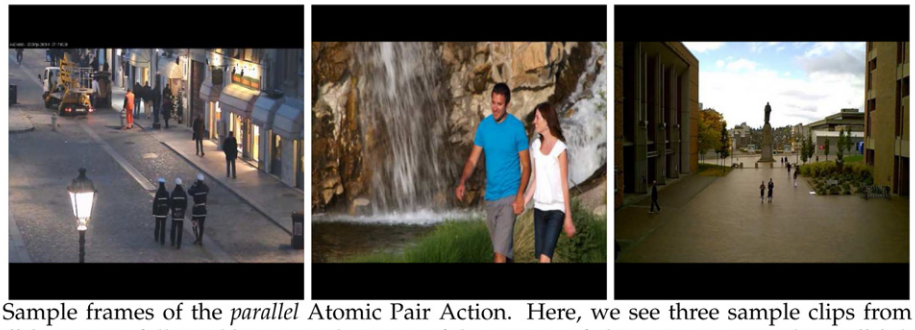
Our approach is to represent multi-step computer vision methods as computational workflows described in high-level declarative notations and capable of processing large quantities of data that comes from multiple sources (Gil et al., 2007a; Taylor et al., 2006). Workflows capture an end-to-end analysis composed of individual analytic steps as a dependency graph that indicates dataflow links as well as control flow links among steps. Our



(a) Sample frames of *converging* Atomic Pair Action. Here, we see three sample clips from the *converging* category, followed by a sample image of the motion of objects in an example *converging* clip.



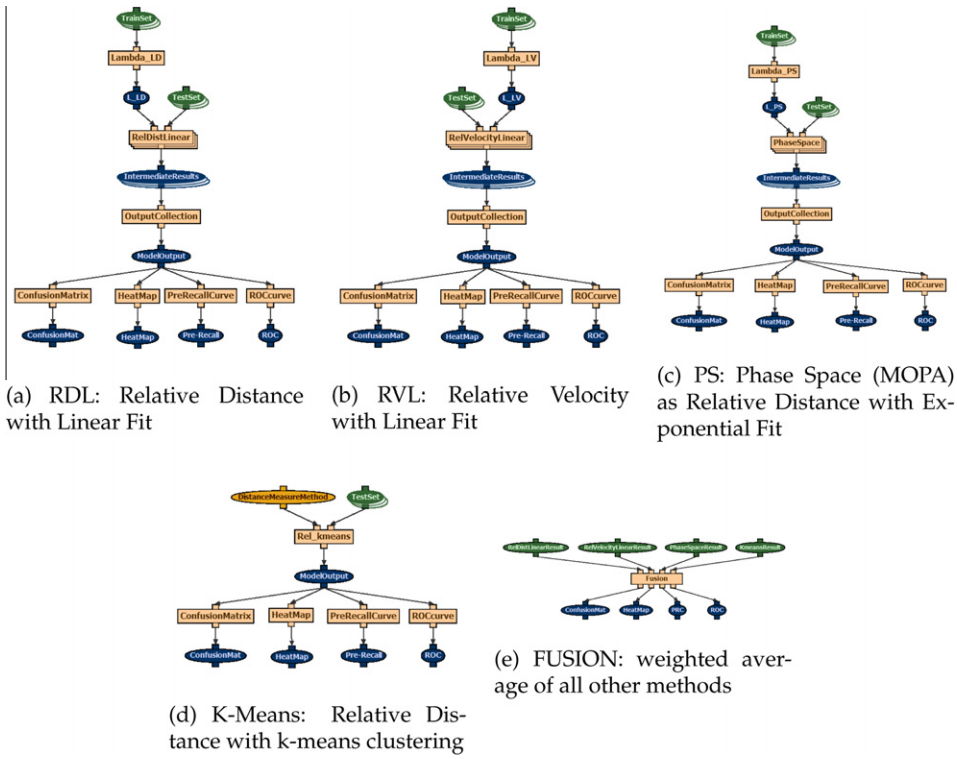
(b) Sample frames of *diverging* Atomic Pair Action. Here, we see three sample clips from the *diverging* category, followed by a sample image of the motion of objects in an example *diverging* clip.



(c) Sample frames of the *parallel* Atomic Pair Action. Here, we see three sample clips from the *parallel* category, followed by a sample image of the motion of objects in an example *parallel* clip.

**Fig. 2.** Sample frames from video clips for all three Atomic Pair Actions categories: *converging*, *diverging*, and *parallel*.

Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015




(a) RDL: Relative Distance with Linear Fit

(b) RVL: Relative Velocity with Linear Fit

(c) PS: Phase Space (MOPA) as Relative Distance with Exponential Fit

(d) K-Means: Relative Distance with k-means clustering

(e) FUSION: weighted average of all other methods

**Fig. 3.** Individual Workflows for the five models described in Section 4.

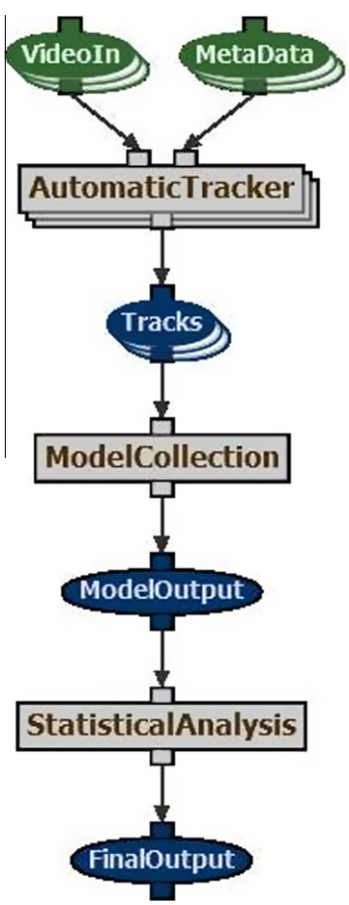



**Fig. 4.** Component Collection: individual workflows integrated as a single Component Collection.

framework uses the Wings workflow system, developed to assist scientists in managing complex computations (Gill et al., 2009, 2011a–d).

A unique feature of Wings is the use of semantic workflows, which represent constraints on the application components within a workflow as well as metadata properties of the data being processed. It has been used in several large-scale distributed scientific applications, including text analytics (Hauder et al., 2011a,b), biomedical image processing (Kurc et al., 2009), earthquake simulations (Gil et al., 2007b), and genomics analysis (Gil et al., 2012). The Wings workflow system is open source and is built on open web standards from the World Wide Web Consortium (W3C). More information and open source software are available at <http://www.wings-workflows.org>.

Workflows effectively capture valuable expertise, as they represent how an expert has designed computational steps and combined them into an end-to- end process. We created a highly reusable family of workflows for video analysis. These workflows capture expertise on using supervised and unsupervised statistical learning algorithms, as they reflect state-of-the-art methods to prepare data, extract features, down-select features, and train models of the data.

This framework provides a common, structured mechanism to compose algorithms and methods for video analysis that is easily extensible to incorporate new algorithms and methods and to process additional types of data. A key objective of this work is to drastically reduce the human effort required to configure and execute new analytic processes from days to minutes by eliminating the need for costly human monitoring and intervention. This requires developing and executing robust end-to-end data analysis processes that include many different algorithms and analytical tools.

Therefore, an important requirement is that the framework needs the ability to easily incorporate new algorithms and new data sources. In addition, the framework is self-adaptive in that it automatically selects methods and parameter values that are

appropriate for the user's data and can analyze its current knowledge, detect missing information, attempt to induce it, and, when this is not possible, assess the value of seeking that missing information (Gil et al., 2011c).

Workflow systems support reproducibility because they capture the details of all workflow execution results as provenance records. Wings publishes workflows and provenance records using the Open Provenance Model (Moreau, 2011), making them available as Linked Open Data (Garijo and Gil, 2011). As a result, it is easy to inspect workflow execution results and reproduce analyses.

## 3. Dataset

The ISI Atomic Pair Actions dataset was collected by recruiting people via Amazon's Mechanical Turk (MTurk) to upload or find public domain videos showing Atomic Pair Actions. Almost all the videos were obtained from a combination of YouTube or public domain datasets. There are three categories of videos showing a single Atomic Pair Action: converging, diverging, or parallel, as shown in Fig. 1. There are 30 videos in each category for a total of 90 videos.

The videos were altered to ensure uniformity and all videos have the following characteristics:

- Cropped to 640 × 480 pixels
- Frame rate of 30 fps
- File that includes the origin URL for provenance
- Image that shows the Ground Truth for that video with bounding boxes around objects of interest, as well as an outline of their approximate trajectory
- Metadata file that includes a ground truth classification, the filename, and the tracks for the objects of interest

Sample clips for each category can be seen in Fig. 2. Each video is a few seconds in length, the shortest is 1 s and the longest is 7 s, with average length of 2–3 s. For each video sequence, we manually cropped the video, converted it as per the characteristics above, and assigned a ground truth Atomic Pair Action category.

We used a tracker based on the particle filter algorithm (Turaga et al., 2008; Nayak et al., 2011) with manual initializations in order to track each objects' motion trajectory.These tracks are included in the metadata file that accompanies each video. In our implementation, we could not track fast motion very well; because of that, the track results became a little bit shorter and resulted in lower results for our models based on relative distance (e.g., for the phase space model this resulted in a lot of noise in training/testing and led to lower accuracy). In addition, the tracking fails quite often due to occlusions, dropped frames, etc.; thus, manual re-initializations were used to re-acquire the tracks. Some clips had perspective motion in the z-direction (into or out of the screen) led to tracking errors, as well. In future work, we will consider a motion segmentation algorithm with automatic re-initializations for automatic tracking (Song et al., 2011).

## 4. Implementation

### 4.1. Models we utilize

In this article, we implement the most appropriate models for the ISI Atomic Pair Actions dataset, derived from Wang et al.

| RDL | c | p | d |
|---|---|---|---|
| c | 0.866667 | 0.133333 | 0.000000 |
| p | 0.000000 | 0.785714 | 0.214286 |
| d | 0.000000 | 0.000000 | 1.000000 |

| RVL | c | p | d |
|---|---|---|---|
| c | 0.133333 | 0.066667 | 0.800000 |
| p | 0.000000 | 0.857143 | 0.142857 |
| d | 0.000000 | 0.428571 | 0.571429 |

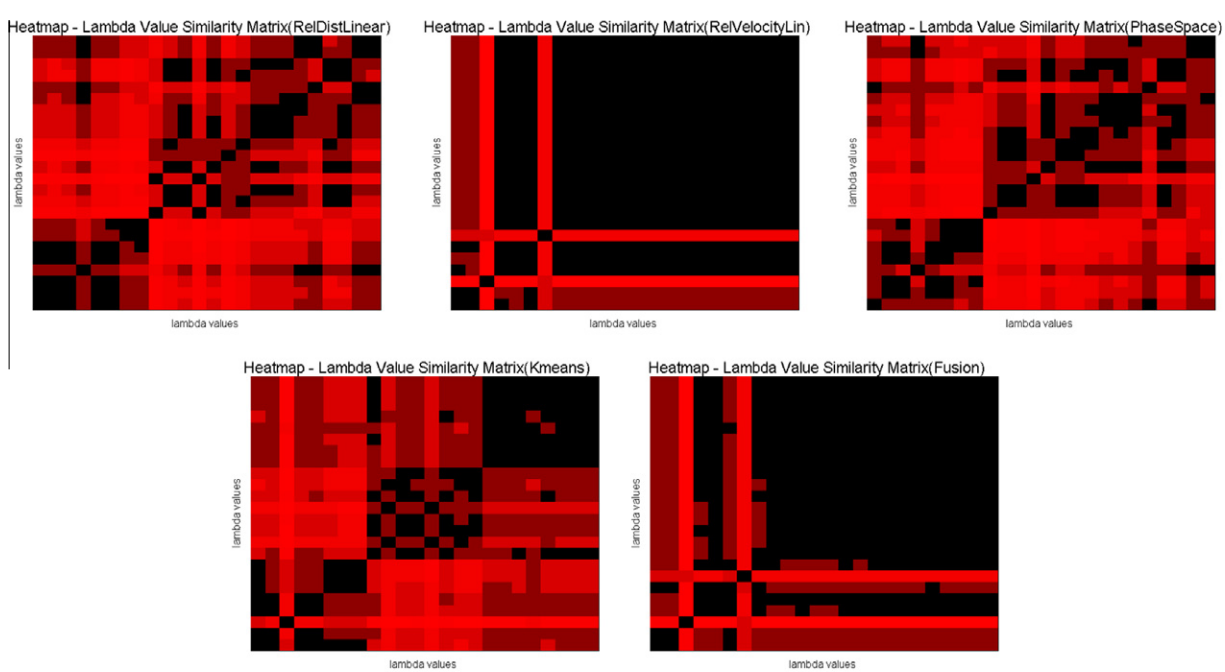
| PS | c | p | d |
|---|---|---|---|
| c | 0.733333 | 0.266667 | 0.000000 |
| p | 0.000000 | 0.642857 | 0.357143 |
| d | 0.000000 | 0.000000 | 1.000000 |

| K-Means | c | p | d |
|---|---|---|---|
| c | 1.000000 | 0.000000 | 0.000000 |
| p | 0.000000 | 1.000000 | 0.000000 |
| d | 0.000000 | 0.125000 | 0.875000 |

| FUSION | c | p | d |
|---|---|---|---|
| c | 0.125000 | 0.875000 | 0.000000 |
| p | 0.000000 | 0.555556 | 0.444444 |
| d | 0.000000 | 0.000000 | 1.000000 |

(a) Confusion Matrices for each of the 5 models

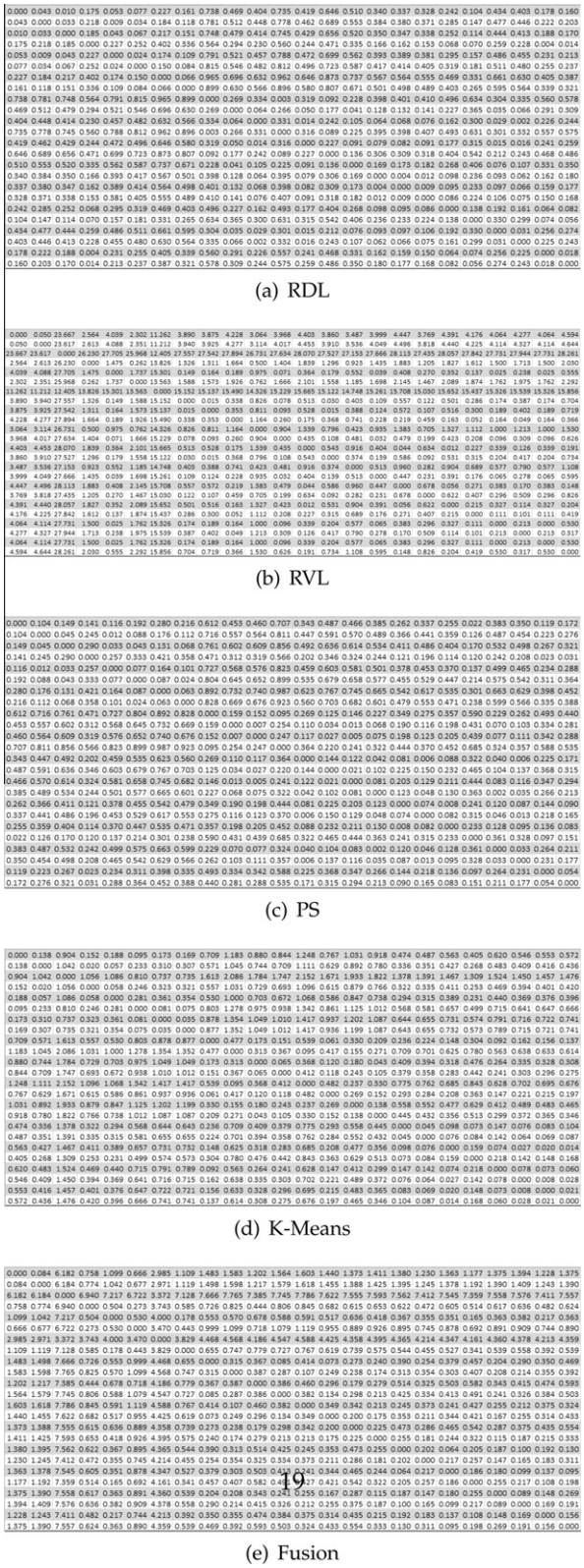
**Fig. 5.** Confusion Matrices for each of the five models described in Section 4 on the ISI Atomic Pair Actions dataset. K-Means outperforms the others.




(a) Heatmaps for each of the 5 models: RDL, RVL, PS, K-Means, and Fusion

**Fig. 6.** Heatmaps for each of the five models described in Section 4 on the ISI Atomic Pair Actions dataset. K-Means outperforms the others while RVL under-performs all others.

Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015



(a) RDL

(b) RVL

(c) PS

(d) K-Means

(e) Fusion

**Fig. 7.** Distance Matrices for each of the five models described in Section 4 on the ISI Atomic Pair Actions dataset.

(2010); Zhang et al. (2006); Sethi and Roy-Chowdhury (2010a,b); Oliver et al. (1999, 2000); Ni et al. (2009) and Zhou et al. (2011), as components in the Wings workflows system; these models are:

1. RDL: in this model, we use the relative distance with a linear fit. We then use the slope, $m$, to do the classification by training the model to find the range of $m$ for *parallel* videos. The minimum of this range determines the maximum of the *converging* videos (slope has to be less than zero) and the maximum of this range determines the minimum of the *diverging* videos (slope has to be greater than zero). We then normalize the scores for each region via $\frac{(score-Min)}{abs(Max)-abs(Min)}$. Finally, in computing the score for ranking the matched videos in each class, we used the distance from the centroid of the range, where $Range = centroid + (abs(Min) + abs(Max)) * .25$ and $centroid = Min + \left(\frac{abs(Min)+abs(Max)}{2}\right)$.
2. RVL: in this model, we use the relative velocity with a linear fit. We used the same process as for RDL, wherein we used a training dataset to find the range of $m$ for *parallel* videos and used the minimum and maximum of this range to determine the maximum and minimum of the *converging* and *diverging* videos, respectively. The score was also found in the same was as for RDL.
3. PS: in this model, we use the Phase Space (MOPA) model implemented as a relative distance with exponential fit model. Here, we used the training videos to train the $\lambda$ of the exponential fit ($e^{-\lambda t}$) for the parallel videos, analogously to the RDL and RVL cases. However, it should be noted that results would be more consistent with the original paper if all three classes were trained on $\lambda$ individually. Here, again, the score was found identically to the method for RDL.
4. K-Means: in this model, we use relative distance with a K-Means clustering. Here, we used a 2-dimensional feature vector, $(\lambda, m)$ , consisting of the features from PS and RDL, respectively. We tested the K-Means clustering with three different distance measures (Euclidean, Manhattan, and Cosine) and Manhattan was the best and so was used in reporting the comparative results. We also needed to use roughly the same number of samples for each of the classes otherwise K-Means was unable to discriminate between the *parallel* and *diverging* cases, unlike the other three (RDL, RVL, and PS) which did not need equal-sized training video sets. Finally, in computing the score used for ranking.
5. FUSION: in this model, we take a weighted score of all of the above methods. Here, we did a majority consensus for the class determination and then took the mean of the scores of the other four models to determine the consensus' score. It should be noted that we could improve the performance of even this simple Fusion method by dropping the score of the models that disagree with the consensus classification.

The individual workflows corresponding to the above five models are shown in Fig. 3 and described further in the next section.

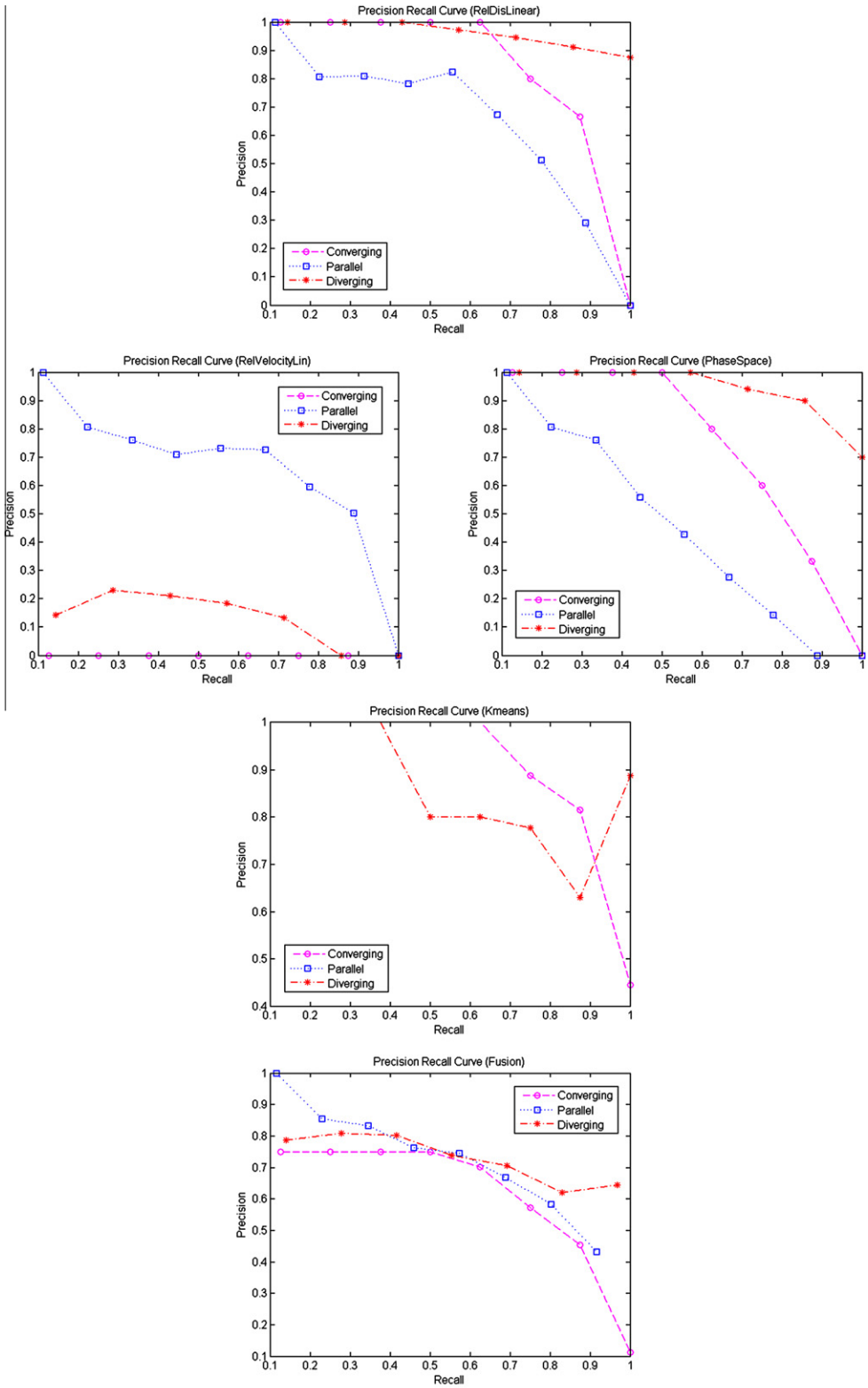
### 4.2. Workflows in Wings

Overall, Wings workflows have the ability to directly incorporate any model in an unsupervised approach, where we use a generative model to fit the data directly. In addition, Wings allows for very simply including a Supervised Learning component by automatically partition the data into a training and testing set, as seen in Fig. 3. In fact, Wings further has the capability to do cross-fold validation, component collections, etc., as detailed next.

The previously defined models were implemented in workflow fragments (see Fig. 3) and can be executed independently from each other. This allowed us to develop each of the workflow fragments separately and distribute development across personnel as



Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015




(a) Precision-Recall curves for each of the 5 models: RDL, RVL, PS, K-Means, and Fusion

**Fig. 8.** Precision-Recall Curves for each of the five models described in Section 4 on the ISI Atomic Pair Actions dataset. PS outperforms the others, although RDL and Fusion are close seconds; RVL is a distant last, however.

needed. The workflow components can be written in heterogenous languages: e.g., some components (Sethi and Roy-Chowdhury, 2009, 2010a–c) are in Java, others in matlab, and still others in C++ but the language of choice is irrelevant as the components are integrated into the workflows without reliance upon their individual implementation idiosyncracies.

Wings has the capability to compare models efficiently through a workflow. A step of the workflow can be a component collection,

which Wings automatically expands to run each of the model components in the collection (Gil et al., 2009), as seen in Fig. 4, where the workflow automatically expands into a workflow that effectively runs the ones shown in Fig. 3(a–c).

We then export (Garijo and Gil, 2011) these workflows and make them available as part of a workflow library so that other researchers can utilize any single component (or the entire component collection) in their own workflows directly by simply importing our components. In addition, these exported workflows can be 75 adapted by adding or changing any component.

In general, when experimenting with Workflows, a user would select a workflow, a dataset, and provide parameter values to run the workflow. Based on the dataset, the configurable parameters, and the number of abstract components, a list of executable workflows is generated. Since abstract components should be specialized within a workflow procedure, the Wings workflow system considers all their possible combinations by using a brute force approach. The Wings semantic reasoner, a part of the workflow system, automatically rejects invalid combinations of components if they violate the constraint rules specified in the data flow. A detailed description of the user interface and the interaction dialogue is described in (Kim et al., 2009).

### 4.3. Variations and limitations

For our Wings setup, we had to ensure the input and output formats were exact and contained all the parameters needed for subsequent components were saved in the intermediate files or parameters. One of the strongest points of Wings was that we can iterate a robust component or task by setting the collection flag variable. When a component gets multiple, heterogenous input files, we have two choices for the module to handle the multiple inputs: doing cross-product of the inputs ($m * n$) or handling them pair-wise ($m + n$). Thus, in the Fusion module, we had to use pair-wise analysis of inputs from the other models; in fact, in all of our components, we explicitly could not do a cross-product of inputs. In addition, we also tried some variations of the workflows; e.g., in K-Means, we tried multiple distance metrics, including Euclidean, Manhattan, and Cosine.

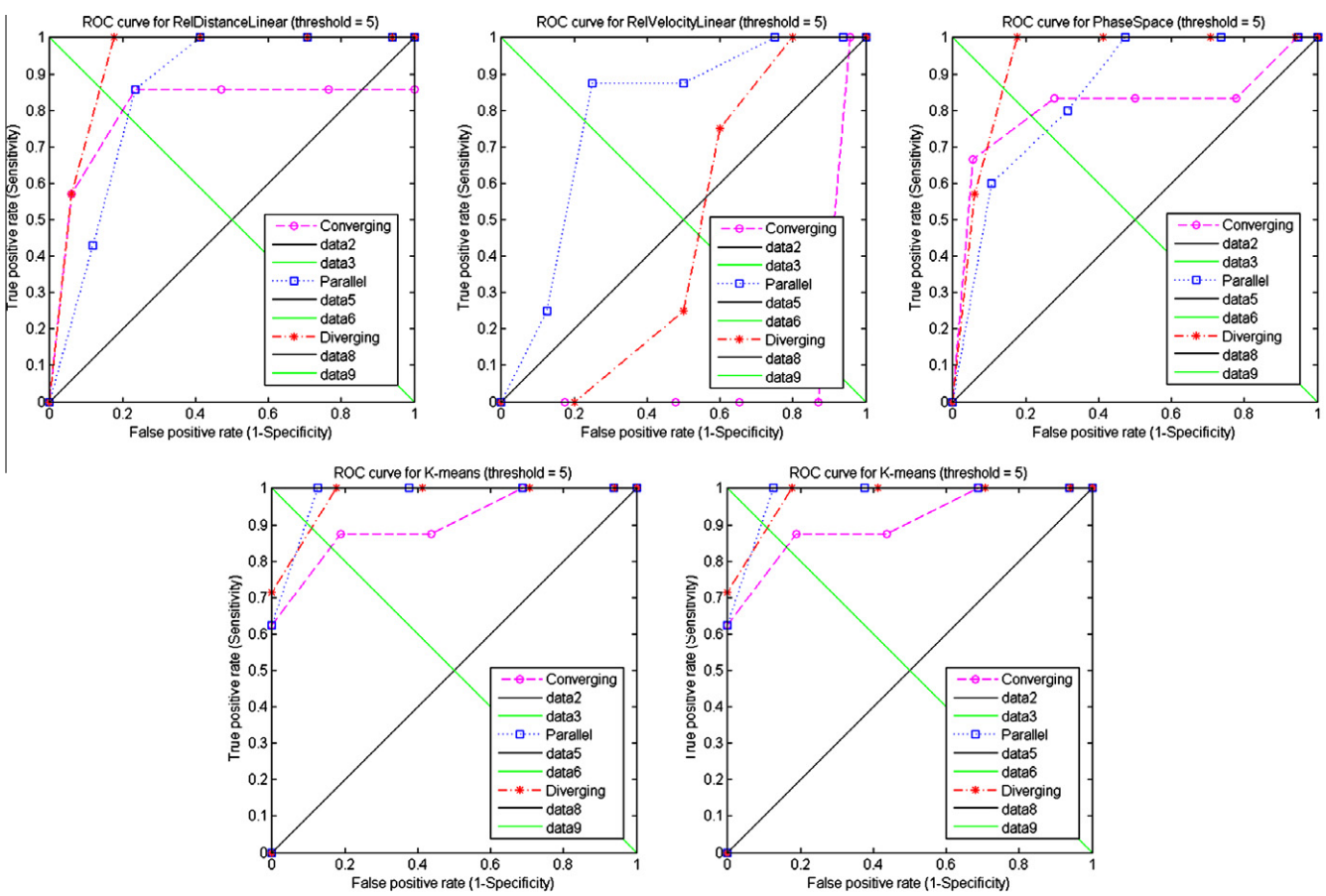
## 5. Evaluation

We provide extensive validation of our structured approach by showing a vast variety of results. We show results for all three classes (*converging*, *diverging*, and *parallel*) on the individual models and provide summary statistics across all the models in the next two sections.

### 5.1. Quantitative Results for Individual Models

In order to show the power of our approach, we calculate a variety of popular statistical measures on each of the five individual models for all three classes (*converging*, *diverging*, and *parallel*); these measures are:

1. Confusion Matrices: these are shown in Fig. 5.
2. Distance Matrices: these are shown in Fig. 7.
3. Heatmaps: these are shown in Fig. 6.




(a) ROC Curves for each of the 5 models: RDL, RVL, PS, K-Means, and Fusion

**Fig. 9.** ROC curves for each of the five models described in Section 4 on the ISI Atomic Pair Actions dataset. K-Means outperforms the others while RVL again under-performs all others.

Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015

**Table 1**
Cumulative Statistical Comparison of all five models and all three categories. Here we see the K-Means outperforms in the F-Measure, EER, and AP/MAP. RVL is the worst performer in all categories.

| | F-Measure | | | Equal Error Rate | | | Average Precision | | | MAP |
|---|---|---|---|---|---|---|---|---|---|---|
| | Converging | Diverging | Parallel | Converging | Diverging | Parallel | Converging | Diverging | Parallel | |
| RDL | 0.928571 | 0.823529 | 0.814815 | 0.235300 | 0.176500 | 0.235300 | 0.875000 | 0.961700 | 0.642500 | 0.826400 |
| RVL | 0.235294 | 0.320000 | 0.800000 | 0.956500 | 0.250000 | 0.600000 | 0.000000 | 0.150900 | 0.672500 | 0.274500 |
| PS | 0.846154 | 0.736842 | 0.666667 | 0.277800 | 0.315800 | 0.176500 | 0.750000 | 0.957100 | 0.422000 | 0.709700 |
| K-Means | 1.000000 | 0.933333 | 0.941176 | 0.187500 | 0.125000 | 0.176500 | 0.930600 | 0.861100 | 1.000000 | 0.930600 |
| FUSION | 0.222222 | 0.777778 | 0.476190 | 0.000000 | 0.235000 | 0.467000 | 0.638900 | 0.640300 | 0.738500 | 0.672600 |

4. ROC curves: the True Positive Rate vs False Positive Rate is shown in Fig. 9.
5. Precision-Recall (PRC) Curves: these are shown Fig. 8.
   The threshold for the PRC curves was set to 0.5 so we looked around the centroid in that range by using $Range = centroid + (abs(Min) + abs(Max)) * .25$ where $centroid = Min + \left(\frac{abs(Min)+abs(Max)}{2}\right)$.

We show the Confusion Matrices, the Distance Matrices, and the heatmaps for each of the five models in Figs. 5, 7, and 6, respectively. We also show the Precision-Recall and ROC Curves for each of the five models in Figs. 8 and 9. According to these results, the K-Means method seems to outperform all, although Fusion is a close second; however, RVL is by far the worst in all instances.

### 5.2. Cumulative and summary statistical comparisons

In addition to the quantitative results on the individual models, we show an extensive comparative analysis using the following standard measures for sets:

1. F-measure: a set-based measure computed on un-ordered sets (directly using precision and recall, also set-based measures) that gives the weighted harmonic mean of precision and recall.
2. Mean Average Precision (MAP) and Average Precision (AP) of Precision-Recall Curves: the MAP is the average of the AP for each category (*converging*, *diverging*, and *parallel*) across each of the five models. This provides a single-figure measure of quality across recall levels.
3. Equal Error Rate (EER) of False Negative Rate vs False Positive Rate: This is calculated as $\frac{FP}{FP+TN}$ where FP is the False Positive Rate and TN is the True Negative Rate. In general, the lower the EER, the higher the accuracy.

Table 1 shows summaries of each of these for all five models and all three categories. According to this table, it shows that K-Means performs best, although the FUSION method is a close second, and RVL is the worst performer in all categories.

### 5.3. Computational speed notes

Another advantage of the Wings workflows framework is that it allows for easy computation of the running times for each of the workflows and components. In general, we found that K-Means was the fastest (clocking in at a running time of 19secs) while Fusion was the slowest, which took almost 18 min because multiple steps were involved, such as collecting output from three different models (thus, it had to execute all of them prior to its processing). RDL, RVL, and PS had to process each track individually as a preliminary step so were intermediate in running time (this is because they had to process all the tracks individually and then collate them for the statistics module, which was a collection component). On average, the algorithms took on average 10–15s, based on the server, current load, network I/O, etc.

## 6. Conclusion

In this article, we presented a novel dataset, the ISI Atomic Pair Actions dataset. In addition, we provided a systematic framework for evaluation of different models that allows easy extension, replication, and comparison of any algorithm or model. Finally, we used the framework on our dataset to provide a comparative analysis of five models for pair-wise action analysis. This analysis showed that the Phase Space (PS) model was most robust but the KMeans and Fusion models performed best under certain circumstances. In future work, we intend to finish implementation of additional models (Gaur et al., 2009; Zhou et al., 2011; Oliver et al., 1999) and also add additional evaluation modules for Cumulative Match Characteristic (CMC or CMScore) curves. In addition, we intend to add in supervised learning components for models like PS in order to fully evaluate the range for its gamma parameter. Finally, we intend to expand the dataset to have additional videos and consider additional scenarios for more complex pair and group actions.

## Acknowledgment

This work was partially supported by the National Science Foundation under Grant #1019343 to the Computing Research Association for the CIFellows Project.

## References

Aggarwal, J.K., Cai, Q., 1999. Human motion analysis: A review. In: CVIU.
Aggarwal, J.K., Ryoo, M.S., 2012. Human activity analysis: A review. ACM Comput. Surv.
Ahad, A.R., 2011. Computer Vision and Action Recognition: A Guide for Image Processing and Computer Vision Community for Action Understanding, Atlantis Ambient and Pervasive Intelligence. Atlantis Press.
Duchenne, O., Laptev, I., Sivic, J., Bach, F., Ponce, J., 2009. Automatic annotation of human actions in video. In: ICCV.
Garijo, Daniel, Gil, Yolanda, 2011. A new approach for publishing workflows: Abstractions, standards, and linked data. In: SC WORKS, Seattle, Washington.
Gaur, Utkarsh, Song, Bi, Roy-Chowdhury, Amit K., 2009. Query-based retrieval of complex activities using strings of motion-words. In: WMVC.
Gil, Yolanda, Deelman, Ewa, Ellisman, Mark H., Fahringer, Thomas, Fox, Geoffrey, Gannon, Dennis, Goble, Carole A., Livny, Miron, Moreau, Luc, Myers, Jim, 2007a. Examining the challenges of scientific workflows. IEEE Comput. 40 (12), 24–32.
Gil, Yolanda, Ratnakar, Varun, Deelman, Ewa, Mehta, Gaurang, Kim, Jihie, 2007b. Wings for pegasus: Creating large-scale scientific applications using semantic representations of computational workflows. In: AAAI, Vancouver, British Columbia, Canada, pp. 1767–1774.
Gil, Yolanda, Groth, Paul, Ratnakar, Varun, Fritz, Christian, 2009. Expressive reusable workflow templates. In: Proc. 5th IEEE Internat. Conf. on e-Science (e-Science), Oxford, UK.
Gil, Yolanda, Ratnakar, Varun, Kim, Jihie, Gonzalez-Calero, Pedro Antonio, Groth, Paul, Moody, Joshua, Deelman, Ewa, 2011a. Wings: Intelligent workflow-based design of computational experiments. IEEE Intell. Systems 26 (1).
Gil, Yolanda, Szekely, Pedro, Villamizar, Sandra, Harmon, Thomas, Ratnakar, Varun, Gupta, Shubham, Muslea, Maria, Silva, Fabio, Knoblock, Craig, 2011b. Mind your metadata: Exploiting semantics for configuration, adaptation, and provenance in scientific workflows. In: Proc. 10th Internat. Semantic Web Conf. (ISWC), Bonn, Germany.

Gil, Yolanda, Szekely, Pedro, Villamizar, Sandra, Harmon, Thomas C., Ratnakar, Varun, Gupta, Shubham, Muslea, Maria, Silva, Fabio, Knoblock, Craig A., 2011c. Mind your metadata: Exploiting semantics for configuration, adaptation, and provenance in scientific workflows. In: ISWC.

Gil, Yolanda, Gonzalez-Calero, Pedro Antonio, Kim, Jihie, Moody, Joshua, Ratnakar, Varun, 2011d. A semantic framework for automatic generation of computational workflows using distributed data and component catalogs. J. Exp. Theor. Artif. Intell. 23(4).

Gil, Yolanda, Ratnakar, Varun, Deelman, Ewa, Mason, Christopher, 2012. Using semantic workflows for genome-scale analysis. In: Internat. Conf. on Intell. Systems Mol. Biol. (ISMB), Long Beach, CA.

Hauder, Matheus, Gil, Yolanda, Liu, Yan, 2011a. A framework for efficient text analytics through automatic configuration and customization of scientific workflows. In: Proc. 7th IEEE Internat. Conf. on e-Science, Stockholm, Sweden.

Hauder, Matheus, Gil, Yolanda, Sethi, Ricky, Liu, Yan, Jo, Hyunjoon, 2011b. Making data analysis expertise broadly accessible through workflows. In: Proc. 6th Workshop on Workflows in Support of Large-Scale Science (WORKS'11), held in conjunction with SC 2011, Seattle, Washington.

Kim, Jihie, Gil, Yolanda, Spraragen, Marc, 2009. Principles for interactive acquisition and validation of workflows. J. Exp. Theor. Artif. Intell.

Kurc, Tahsin M., Hastings, Shannon, Kumar, Vijay S., Langella, Stephen, Sharma, Ashish, Pan, Tony, Oster, Scott, Ervin, David, Permar, Justin, Narayanan, Sivaramakrishnan, Gil, Yolanda, Deelman, Ewa, Hall, Mary W., Saltz, Joel H., 2009. High performance computing and grid computing for integrative biomedical research. J. High Perform. Comput. Appl. 23 (3), 252–264.

Moreau, Luc, Clifford, Ben, Freire, Juliana, Futrelle, Joe, Gil, Yolanda, Groth, Paul, Kwasnikowska, Natalia, Miles, Simon , Missier, Paolo, Myers, Jim, Plale, Beth, Simmhan, Yogesh , Stephan, Eric, Van denBussche, Jan, 2011. The open provenance model core specification (v1.1). Future Gener. Comp. Syst. 27(6).

Nascimento, J.C., Figueiredo, M.A.T., Marques, J.S, 2005. Segmentation and classification of human activities. In: BMVC Workshop on Human Activity Recognition and Modelling.

Nayak, Nandita M., Sethi, Ricky J., Song, Bi, Roy-Chowdhury, Amit K., 2011. Motion pattern analysis for event and behavior recognition. In: Moeslund, T.B., Sigal, L., Krüger, V., Hilton, A. (Eds.), Visual Analysis of Humans. Springer-Verlag, pp. 289–309.

Ni, Bingbing, Yan, Shuicheng, Kassim, Ashraf, 2009. Recognizing human group activities with localized causalities. In: CVPR, pp. 1470–1477.

Oliver, Nuria, Rosario, Barbara, Pentland, Alex P., 1999. A Bayesian computer vision system for modeling human interactions. ICVS. Springer-Verlag.

Oliver, Nuria M., Rosario, Barbara, Pentland, Alex P., 2000. A Bayesian computer vision system for modeling human interactions. PAMI 22 (8), 831–843.

Ryoo, M.S., Aggarwal, J.K., 2009. Stochastic representation and recognition of high-level group activities. In: CVPR.

Sethi , Ricky J., Roy-Chowdhury, Amit K., Ali, Saad, 2009. Activity Recognition by Integrating the Physics of Motion with a Neuromorphic Model of Perception. In: WMVC.

Sethi, Ricky J., Roy-Chowdhury, Amit K., 2010a. Modeling and recognition of complex multi-person interactions in video. In: ACM MM MPVA, pp. 0–3.

Sethi, Ricky J., Roy-Chowdhury, Amit K. , 2010b. Physics-based activity modelling in phase space. In: ICVGIP.

Sethi, Ricky J., Roy-Chowdhury, Amit K., Veeraraghavan, Ashok, 2010c. Gait Recognition Using Motion Physics in a Neuromorphic Computing Frame-work. In: Bhanu, B., Govindaraju, V.,(Eds.), Multibiometrics for Human Identification. Cambridge University Press.

Sethi, Ricky J., Bry, Lynn, The Madsci Network. Technical Report, madsci network research, 2011.

Song, Bi, Sethi, Ricky J., Roy-Chowdhury, Amit K., 2011. Robust wide area tracking in single and multiple views. In: Moeslund, T.B., Sigal, L., Krüger, V., Hilton, A. (Eds.), Visual Analysis of Humans. Springer-Verlag, pp. 1–18.

Taylor, Ian J., Deelman, Ewa, Gannon, Dennis B., Shields, Matthew, 2006. Workflows for e-Science: Scientific Workflows for Grids. Springer-Verlag.

Turaga, P., Chellappa, R., Subrahmanian, V.S., Udrea, O., 2008. Machine recognition of human activities: A survey. In: CSVT.

Vandewalle, Patrick, Kovacevic, Jelena, Vetterli, Martin, 2009. Reproducible research in signal processing. IEEE Signal Process. Mag. 26, 37–47.

Wang, Yaowei, Tian, Yonghong, Duan, Lingyu, Hu, Zhipeng, Jia, Guochen, 2010. ESUR: A system for Events detection in SURveillance video. In: ICIP. IEEE, pp. 2317–2320, number 60973055.

Zhang, Dong, Gatica-Perez, Daniel, Bengio, Samy, 2006. Modeling individual and group actions in meetings with layered HMMs. IEEE Trans. Multimedia 8 (3), 509–520.

Zhou, Yue, Yan, Shuicheng, Huang, Thomas S., 2008. Pair-activity classification by bi-trajectories analysis. In: CVPR.

Zhou, Yue, Ni, Bingbing, Yan, Shuicheng, Huang, Thomas S., 2011. Recognizing pair-activities by causality analysis. ACM Trans. Intell. Systems Tech. 2 (1), 1–20.

Please cite this article in press as: Sethi, R.J., et al. Structured analysis of the ISI Atomic Pair Actions dataset using workflows. Pattern Recognition Lett. (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.015