Proceedings of the 2009 IEEE
International Conference on Information and Automation
June 22 -25, 2009, Zhuhai/Macau, China

Mechanical Feature Attributes for Modeling and Pattern Classification of Physical Activities

Theodoros Theodoridis, Alexandros Agapitos, Huosheng Hu, and Simon M. Lucas
Department of Computer Science, University of Essex
Wivenhoe Park, Colchester CO4 3SQ, U.K.
{ttheod, aagapi, hhu, sml}@essex.ac.uk

***Abstract*— A rigorous investigation on the synergy of mechanical attributes to engineer tactics for measuring human activity in terms of forces, as well as to provide independency and discrimination clarity of action recognition using linear and non-linear classification methodologies from data mining and evolutionary computation, are the main objectives where this paper focuses on. Mechanical analysis is employed to mathematically describe and model human movement by using a number of mechanical features inspired mainly from Kinematics Dynamics. Such features employ a twofold role on the descriptive analysis of an activity, initially to provide statistics regarding inertial expressions, probable hazard levels, body-status of energy loss, and finally to exploit these attributes by decomposing the 3D time series data for pattern recognition in terms of actions and behaviours. The performance statistics are being utilized by a mobile robot for remote surveillance within a smart environment.**

I. Introduction

On the study of mechanical features to examine behavioural activities and analyze pattern occurrences using 3D kinematic models, there is no similar work in the field of data mining in such way where mechanical attributes to provide statistics assessing the physical status of a movement along with classification evaluations of a recognized expression. The physical activity analysis includes action A = {standing[1], handshaking[2], waving[3], clapping[4], pushing[5], pulling[6], slapping[7], punching[8]} and behaviour B = {normal[1], aggressive[2]} recognition respectively.

Previously, in [1] we have shown two different classifier representations based on dynamic Artificial Neural Networks (ANNs) and Genetic Programming (GP) on a performance comparison to recognize physical activities of 3D spatial time series data. More specifically, utilization of evolutionary statistical feature extraction employed non-mechanical features which have actually been used within expression tree structures to model program output distributions for examples of different classes. In a further feature-based modeling analysis and pattern classification [2] we have presented a fusion of a multi auto-adjusted TSK fuzzy logic classifier, used for action recognition, and a signal convolver classifier, used for behaviour discrimination, both employed to model physical actions. The innovative fuzzy-convolution model has used a vast number of feature attributes utilized as configuration parameters to optimize the model's classification performance. In [3], a kinematics-dynamics approach has shown how to estimate joint moments from EMG signals using both forward and inverse dynamics of a neuromusculoskeletal model by accounting force-length and force-velocity relationships so that to estimate muscle forces, joint moments and kinematics from neural signals. Moreover, previous researchers have tackled the problem of classifying a set of examples by inducing decision trees that described the underlying structure of similar dataset. Similar to [4], we allow our GP [5] system to search the space populated by programs composed of arbitrary combinations of `If` statements. Decision tree algorithms such as ID3, CART, OC1 and C4.5 are greedy local search algorithms, thus, By distilling a decision tree down to its behavioural core a set of disjunctive rules are being revealed in a programming language that allows `If` statements and predicates.

Fig. 1 illustrates the hardware configuration setting of the system architecture. A person is shown to act in a 3D intelligent environment performing some physical activity. Two external devices, a 3D tracker (Vicon system) and a mobile robot (SCITOS G5), cooperate as a perception to action unit to produce classification and mechanical attribute-based statistics of the performed activity [1], [2]. The mechanical statistics are divided into three semantic categories representing inertial expressions such as forces applied, hazard levels which might have caused physical assaults, and the actor's body-status in terms of energy lost.

The rest of the paper is organized as follows: In section II, mechanical analysis of human movement describes the speed deployment of an action performed under an inverse dynamics model. Section III presents six linear and non-linear classifier representations. Experimental results are shown in section IV whereas section V points out conclusions and future work.

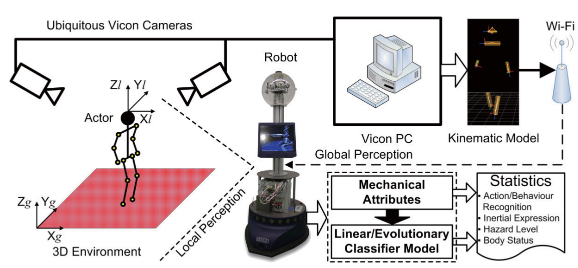

Fig. 1. Configuration setting of a human actor, a mobile robot, and the Vicon 3D tracker.

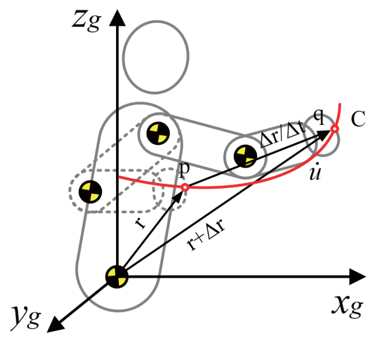

Fig. 2. Description of speed vector on a 3D inertial system of coordinates.

II. Mechanical Analysis Of Human Movement

The mechanical movement of a human limb can be rigorous in its description of how an actor performs some physical activity within an inertial system of coordinates. Fig. 2 illustrates this notion where under the expression of a physical activity, the major speed feature vector $\dot{u}$ is shown to be produced from the trajectory $\mathbf{C}$ through the distance Δr over the time Δt.

Similar to the theoretical vector analysis of [6], in Fig. 2, during the punching activity we observe that the actor's fist is displaced through the curve $\mathbf{C}$ which is the trajectory of a punch. The position vector of point p at time t is $\mathbf{r} = \mathbf{r}(t)$, while the position vector of point q at time $t+\Delta t$ is $\mathbf{r}+\Delta\mathbf{r} = \mathbf{r}(t+\Delta t)$. The speed of the fist at p is given by Equation 1:

$$\mathbf{v} = \frac{d\mathbf{r}}{dt} = \lim_{\Delta t \to 0} \frac{\Delta \mathbf{r}}{\Delta t} = \lim_{\Delta t \to 0} \frac{\mathbf{r}(t+\Delta t) - \mathbf{r}(t)}{\Delta t} \tag{1}$$

which is a vector of p in $\mathbf{C}$. If $\mathbf{r} = \mathbf{r}(t) = x(t)\mathbf{i} + y(t)\mathbf{j} + z(t)\mathbf{k}$ are the right vector components in coordinates x, y, and z, given by Equation 1 while the speed is given by Equation 3:

$$\mathbf{v} = \frac{d\mathbf{r}}{dt} = \frac{dx}{dt}\mathbf{i} + \frac{dy}{dt}\mathbf{j} + \frac{dz}{dt}\mathbf{k} \tag{2}$$

$$u = |\mathbf{v}| = \left|\frac{d\mathbf{r}}{dt}\right| = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2}$$
$$= \frac{ds}{dt} = \dot{u} \tag{3}$$

The speed vector $\dot{u}$ is regarded as the major mechanical attribute of this analysis because it is the base feature used by all the other feature categories shown on Table I.

Inverse dynamics analysis is particularly useful to employ an interface capable to implement a general system model from which physical activities express instantaneous behaviours by measuring the positions, and then estimating a number of mechanical parameters; similar to [3] who they used an inverse dynamics model to estimate forces through position data. Fig. 3 depicts an inverse dynamics model which interfaces 3D position activity information into mechanical attributes through the mechanical feature index of Table I where the equations of the mechanical features are contained. The inverse dynamics model uses a Trajectory Model which incorporates 3D trajectories generated by the human actor. The trajectory model generates 3D coordinates which are position data to be translated in speeds and accelerations and then converted by the Mechanical Feature Index in mechanical forces.

III. Pattern Classification Methods

■ Linear Classification Models

Four feature/error-based linear classification models [7] based on machine learning concepts with innovative internal configurations have been used for the action and behaviour recognition task using mechanical features for pattern discrimination and dimensionality reduction of 3D data time series. Besides the dimensionality reduction, mechanical features are also used to filter noise-corrupted chunks of a segment achieving thus higher classification accuracy by making the classifiers more discriminant as discussed by [2].

The four classifier architectures are illustrated in Fig. 4 where the source which provides information, generated in time series, is shown on the top of each diagram consisting of the *TS Data*, *Signal Decomposition*, and the *Training(TRN)/Testing(TST) Separator* module as described analytically by [2] with training and testing percentages of 70% and 30% respectively. Three more common modules are being employed by the four classification models: (a) time-domain *feature* module, used to reform the training and the testing data into singular statistical feature values, (b) *error-distance* module, used to compute the distance between training and testing data, (c) minimum-average error modules, to select the *minimum-average* error value which is likely to denote the activity being tested.

a. Distance Classifier (*dis*) The simple distance classifier (Fig. 4(a)), exploits the *Error Propagation* (*erp*) attribute to extract a singular feature value for all the training and

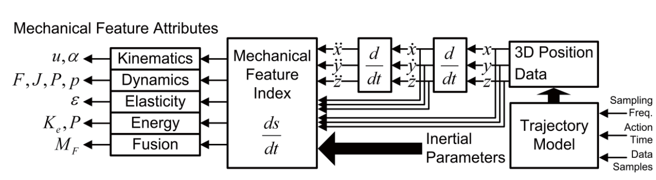

Fig. 3. Inverse dynamics model of human movement.

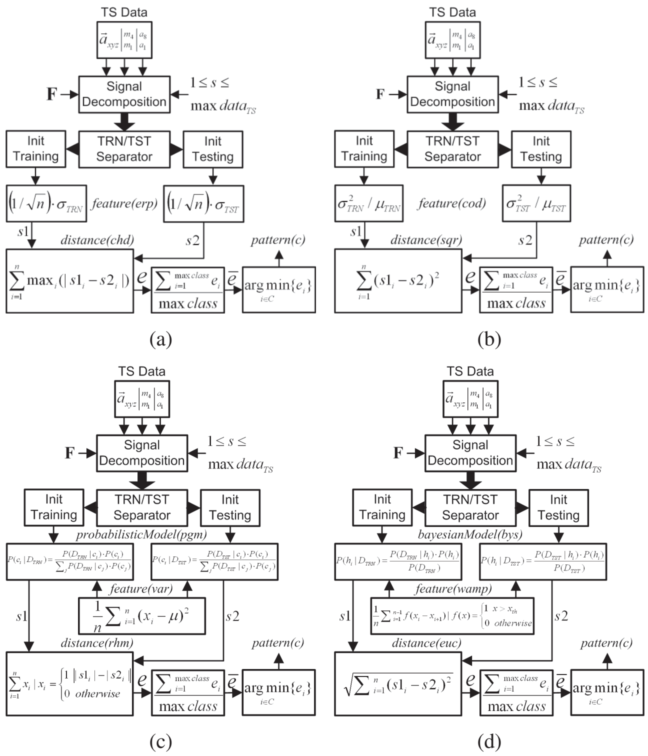

Fig. 4. Classifier architectures, (a) Distance, (b) Statistical, (c) Probabilistic, (d) Bayesian.

the testing data, and then through the *Chebyshev* distance (*chd*) to get the recognized class after being estimating the minimum average error.

b. Statistical Classifier (*sts*) Similar to distance, the *sts* classifier (Fig. 4(b)) uses as a feature the statistical function *Coefficient of Dispersion* (*cod*) to provide a singular statistical evaluation regarding the dispersion of a data set revealing whether a cluster of certain density has been formed. The cluster-based dispersion results are then being checked by the *Square Error* distance.

c. Probabilistic Generative Model Classifier (*pgm*) The *pgm* classifier (Fig. 4(c)) has a modified internal configuration, described by Equation 4, where the *Variance* feature supplies the model with variational estimates of the data. The training/testing probabilities are thereafter passed for error generation using the *Rounded Hamming* distance.

$$P(c_i|D) = \frac{\frac{D_i(1/n\sum_{j=1}^{n}(x_j-\mu)^2)_{var}}{\sum_j D_{ij}} \cdot \frac{1}{C_{max}}}{\sum_{k=1}^{C_{max}}\left[\frac{D_k(1/n\sum_{j=1}^{n}(x_j-\mu)^2)_{var}}{\sum_j D_{kj}} \cdot \frac{1}{C_{max}}\right]} \tag{4}$$

where c_i is the class being tested, D_i is the data time series of class i, j denotes the number of the individual samples of

TABLE I

MECHANICAL FEATURE INDEX

F	C	Name	Formula	Unit
1	Kinematics	Speed(SPD)	$u = \lvert \frac{s}{t} \rvert$	m/s
2		Acceleration(XLR)	$\alpha = \frac{u(t)-u_0(t)}{\Delta t}$	m/s^2
3	Dynamics	Force(FRC)	$F = m\alpha$	$N \rightarrow kgm/s^2$
4		Impulse(IMP)	$J = F\tau$	$Ns \rightarrow kgm/s$
5		Momentum(MNT)	$P = mu$	$Ns \rightarrow kgm/s$
6		Pressure(PRS)	$p = \frac{F}{A}$	$Nm^{-2} \rightarrow kgm^{-1}s^{-2}$
7	Elasticity	Strain(STR)	$\varepsilon = \frac{s}{\bar{s}_{agg}}$	*Dimensionless*
8	Energy	Kinetic Energy(KEN)	$K_e = \frac{1}{2}m(u^2(t)-u_0^2(t))$	$J \rightarrow Nm \rightarrow kgm^2/s^2$
9		Power(POW)	$P = Fu$	$W \rightarrow kgm^2s^{-3}$
10	Fusion	MechFusion(MFU)	$M_F = F + (u \cdot s) + (K_e \cdot \varepsilon)$	*Dimensionless*

Indicators: 'τ' trajectory action time, 'A' striking area, '$\bar{s}_{agg}$' average aggressive speed times the τ.

a data time series, C_{max} is the maximum number of classes, x represents the individual data samples, and μ is the mean.

d. Bayesian Classifier (*bys*) Similar to *pgm*, the *bys* classifier (Fig. 4(d)) also generates pattern probabilities. The *Willison Amplitude* feature supplies the model with a number of counts for each change of a signal amplitude that exceeds a predefined threshold (Equation 5). The thresholded values are thereafter passed to the *Euclidean* distance to be checked.

$$P(h_i|D) = \frac{\frac{D_i(1/n\sum_{j=1}^{n-1}f(x_j-x_{j+1}))_{wamp}}{\sum_j D_{ij}} \cdot \frac{1}{C_{max}}}{\sum_j D_{ij}} \tag{5}$$

where $f(x) = \begin{cases} 1 & x > x_{th} \\ 0 & \text{otherwise} \end{cases}$.

■ Binary, Linear/Non-Linear Decision Trees

Binary is the classical form of a decision tree induced by GP in which each predicate is a disjunction or conjunction of conditions on a single attribute (i.e. $x > 0.4$). A *linear* decision tree is one in which each condition contains a linear combination of one or more attributes which is tested against a constant value (i.e. $3x + y > 0.1$) whereas a *non-linear* decision tree allows non-linear combinations of attributes (i.e. $3x^2 + y^3 > 0.7$).

For evolutionary algorithm we used a panmictic, generational genetic algorithm combined with elitism (1%). The algorithm uses tournament selection with a tournament size of 20. In order to allow for more exploitation towards the end of each evolutionary run, the tournament size has been made dynamic during the final 15 generations incremented by a percentage of 8% in each generation. The evolutionary run proceeds for 100 generations and the population size is set to 5000 individuals. Evolution halts when all of 100 generations have elapsed. Ramped-half-and-half tree creation with a maximum depth of 8 is used to perform a random sampling of program space during the initial generation.

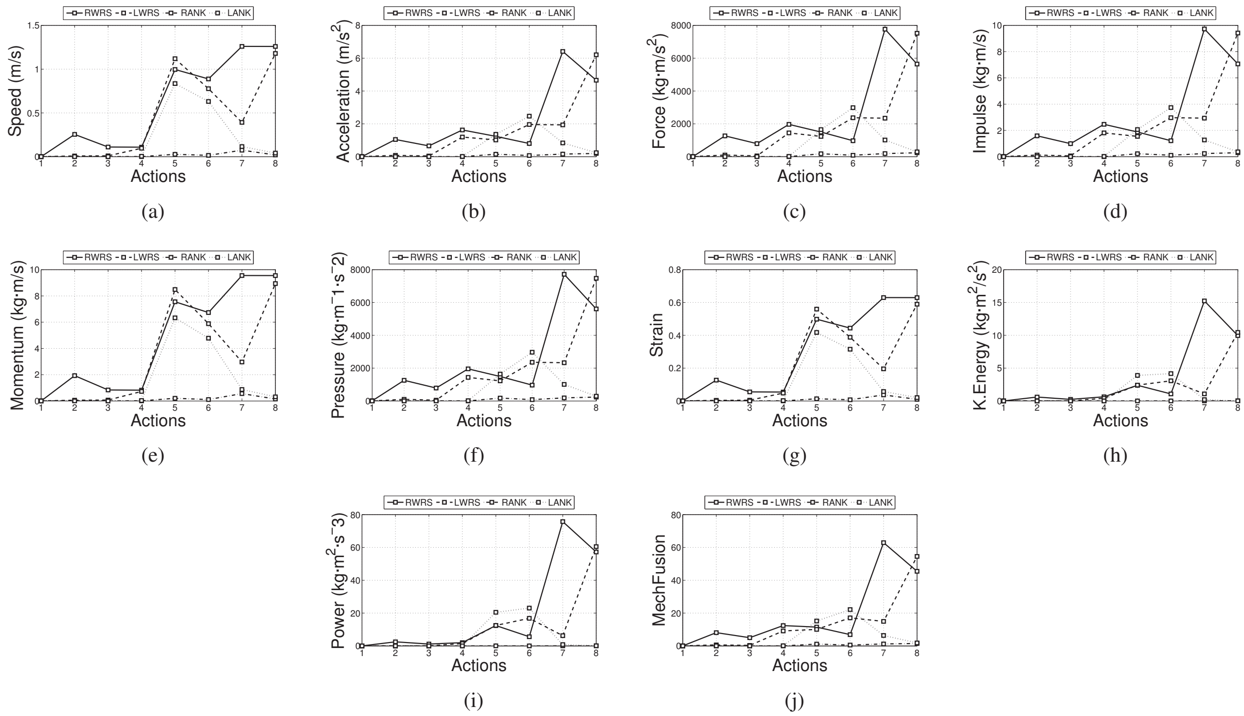

Fig. 5. Physical activities expressed by mechanical features using four markers attached on the limbs, (a) SPD, (b) XLR, (c) FRC, (d) IMP, (e) MNT, (f) PRS, (g) STR, (h) KEN, (i) POW, (j) MFU.

During the run, expression trees are allowed to grow up to depth of 17. The fitness function is based on *classification accuracy* which is induced by dividing the number of correct classifications by the number of training cases in the learning dataset. Heuristic search employs a mixture of mutation-based variation operators [8].

IV. Experimental Results

Before attempting to classify the pattern activities it is necessary to examine how mechanical features represent each individual action and the corresponded behaviour. In all subfigures of Fig. 5, marker RWRS (right wrist) seems to be more active than any other marker and it is the one to provide the most significant information to all classifiers for action recognition since the magnitude in each action, for all the mechanical features, sustains a particular level. Marker LWRS (left wrist) has also similar behaviour and it is also appropriate to be used for action classification. On the contrary, the leg markers RANK (right ankle) and LANK (left ankle), appear less significant behaviour in terms of discrimination since they bring out some actions with sporadic and poor indications for each behaviour; a fact which makes the leg markers inappropriate for classification especially the RANK marker which shows lethargic performance. From the subfigures, it is hence visible that the arms and the left leg have been used the most for all the actions. Thereby, the mechanical feature categories which contribute the most to draw patterns are the Kinematics, the Dynamics, and the Energy with dominant the Kinematics category.

Assessing the overall performance, we conclude that the actor moved with speed and acceleration 1.25m/s and 4.65m/s^2 respectively by using force up to 5642.30N. The momentum of the arm moved with 9.54Ns so that to explode with impulse 7.06Ns. The impulse created pressure of 5607.00Nm^{-2} via the area of the fist, where the shocking power of the fist to the target was 57.22W. The overall energy spent regarding all the markers of the punching activity was 20.44J. Thus, from this physical performance taken from a particular dataset, the actor's rapid physical expression is not hazardous but it can still cause injuries or destroy property.

A. Classification of Physical Activity

1) Linear Classifiers: Due to the fixed architectures and the static behaviour of the linear classification models, 40 experimental runs have been carried out one for each classifier over the ten mechanical features. Primarily, we are interested

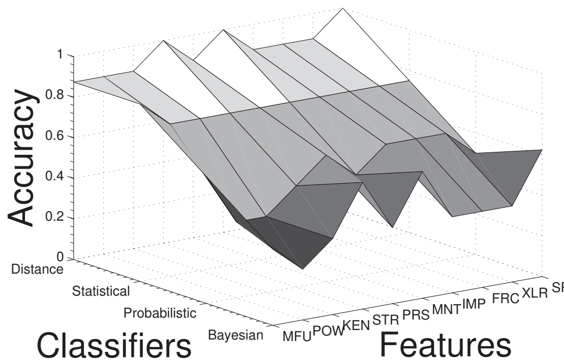

Fig. 7. Classification accuracy performance, features versus linear classifier models.

to investigate the mechanical features which led the the four independed classifiers to achieve distinct performances. Fig. 7 depicts a summary performance of the four classifiers over the ten mechanical features. In a closer look, we observe that all the distinct features come from different categories and more than one linear classifier has taken advantage of them; these features are: *speed* (SPD), *momentum* (MNT), *strain* (STR), and *mech-fusion* (MFU).

2) Evolutionary Decision Trees: Two different representation spaces a) binary; b) linear/non-linear decision trees are being studied in a sequence of 20 independent evolutionary runs for each one and statistics are being reported. Figures 6(a), 6(d) present the learning curves under the binary and linear/non-linear representations respectively. We observe that on average both representations reach moderate accuracy (average curve in bold), as measured by the percentage of the training cases correctly classified. However, the learner under the binary representation seems to learn faster as evidenced by the reach of an accuracy of 0.5 by generation 28 as opposed to generation 32 of linear/non-linear representation and an accuracy of 0.6 by generation 80 as opposed to generation 92.

The evolution of use of mechanical features forming the conditions that determine the class of a pattern is depicted in Figures 6(b) and 6(e) for binary and linear/non-linear decision trees respectively. The axis labeled "Mechanical Features" has been decomposed in 12 parts each representing one input time series. Each such part is further decomposed in 10 points each representing the speed, momentum, acceleration, impulse, force, kinetic energy, power, pressure, strain, and mech-fusion extracted from the respective marker time series. Labeling these on the axis has been omitted for clarity. Figures 6(c) and 6(f) present mean of features' existence within the population at the end of the evolutionary run, averaged over 20 independent trials.

For binary decision trees we observe that mechanical features have been mainly extracted from the markers positioned on the left ankle and left and right wrists (induced by setting a threshold of the feature average usage to 0.5) as Fig. 5 verifies. From these features, the dominant ones (threshold of 0.6) appear to be: (a) momentum and acceleration of left ankle; (b) speed, momentum and pressure of left wrist; (c) speed and strain of right wrist. For linear/non-linear decision trees we observe a similar tendency of not significant use of the mechanical features extracted from the right ankle (usage threshold set to 1.4). Dominant features include: (a) kinetic energy of left ankle; (b) momentum and pressure of left wrist; (c) acceleration of right wrist.

3) Performance Measures: The final step of the experimental analysis is to examine the accuracy of the eight individual actions achieved by the six classifiers as it is shown in Fig. 8. It is statistically visible that the classifiers accomplished exceptional accuracy on the normal behavioural group of actions 1-4, whereas for the aggressive behavioural group, actions 5-8 the accuracy drops with recognition error which appears to be higher as the error bars of Fig. 8 show. The linear classifiers (Fig. 8(a), 8(b), 8(c), and 8(d)) have shown handfoul performance on the *handshaking* as well as on the *punching* action while the accuracy of the rest actions was satisfactory. For the evolutionary decision trees, Fig. 8(e) and 8(f) illustrate the average classification accuracy on the test set achieved in each of the classes for binary and linear/non-linear representation respectively. We can observe that both representations achieved moderate performance. Also, both classifier representations exhibit similar difficulty on identifying *pulling*, *slapping* and *punching*. Ultimately, Table II demonstrates the performance results with propelling the distance classifier.

V. Conclusions And Future Work

Ten mechanical features have been employed for modeling physical activities as well as for classifying normal and aggressive patterns by six classifier representations of linear and non-linear data mining and evolutionary-based architectures. The experimental results have shown common

TABLE II

Performance measures of the six classifier representations.

Measure	**DIS**	**STS**	**PGM**	**BYS**	**DT_1**	**DT_2**
Precision	0.86	0.64	0.41	0.25	0.33	0.32
Recall	0.91	0.76	0.53	0.43	0.40	0.41
Accuracy	0.90	0.76	0.53	0.43	0.39	0.41

Indicators: 'DT_1' Binary Decision Tree 'DT_2' Linear/Non-Linear Decision Tree classifier.

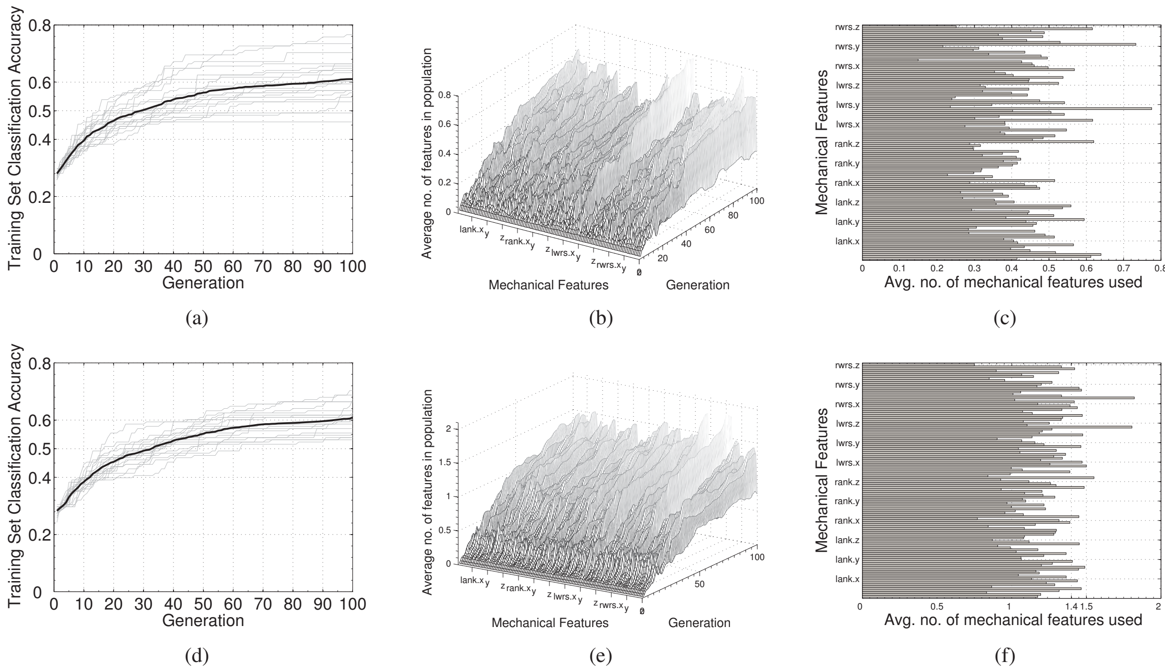

Fig. 6. (a)(b)(c) - Binary decision trees, (d)(e)(f) - Linear/non-linear decision trees. (a,d) Learning Curves, (b,e) Evolution of mechanical features usage, (c,f) Average usage of mechanical features.

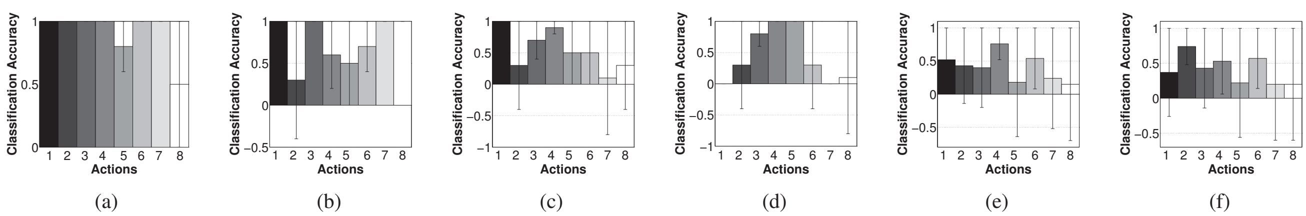

Fig. 8. Classification accuracies of error-based representations (a) Distance, (b) Statistical, (c) Probabilistic, (d) Bayesian, (e) Binary DT, (f) Linear/non-linear DT.

outcomes relating the usage of the most frequent markers, the most preferable features, and the classification accuracy achieved in each of the eight individual actions. In other words, the classifiers benefit the most from the left ankle and left and right wrist markers whereas tendency for usage of common features has been shown on the speed, momentum, pressure, and strain. Beyond the standard mechanical features employed in this work, a plausible future application would be to work with new evolvable features which can inspire inherited attributes from the standard ones as seen in this framework.

REFERENCES

[1] T. Theodoridis, A. Agapitos, H. Hu, and S. M. Lucas. Ubiquitous robotics in physical human action recognition: A comparison between dynamic anns and gp. In *2008 IEEE Int. Conf. on Robotics and Automation*, pages 3064–3069, 2008.

[2] T. Theodoridis and H. Hu. A fuzzy-convolution model for physical action and behaviour pattern recognition of 3d time series. In *Proc. of the 2008 IEEE Int. Con. on Robotics and Biomimetics*, 2008.

[3] T. S. Buchanan, D. G. Lloyd, K. Manal, and T. F. Besier. Neuromusculoskeletal modeling: Estimation of muscle forces and joint moments and movements from measurements of neural command. *Applied Biomechanics*, 20:367–395, 2004.

[4] M. C. J. Bot and W. B. Langdon. Application of genetic programming to induction of linear classification trees. volume 1802 of *LNCS*, pages 247–258, Edinburgh, 2000. Springer-Verlag.

[5] J. R. Koza. *Genetic Programming: on the programming of computers by means of natural selection*. MIT Press, Cambridge, MA, 1992.

[6] M. R. Spiegel. *Theoretical Mechanics*. Schaum's Outline. McGraw-Hill Inc, 1980.

[7] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, Cambridge, UK, 2006.

[8] K. Chellapilla. Evolving computer programs without subtree crossover. *IEEE Trans. on Evolutionary Computation*, 1:209–216, 1997.