UNIVERSITY OF CALIFORNIA
RIVERSIDE

A Physics-Based, Neurobiologically-Inspired Stochastic Framework for Activity
Recognition

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Ricky Jaineet Sethi

August 2009

Dissertation Committee:

    Professor Amit K. Roy-Chowdhury, Chairperson
    Professor Eamonn Keogh
    Professor Christian Shelton

The Dissertation of Ricky Jaineet Sethi is approved:

_____

_____

_____

Committee Chairperson

University of California, Riverside

Acknowledgments

All things in life, despite our best efforts to imagine otherwise, involve luck. We might call it different things: providence, destiny, help from above; but it all boils down to happenstance that, in some fortuitious situations, rules in our favour. Many, many aspects of my research and dissertation reflect this. But luck, like hard work and intelligence, is but one component amongst many. The many, in this case, are all the people that helped make this work what it is. Words fail in expressing the full extent of their influence on my work and, even more, on me, and so I say to all of them a very heartfelt "Thank you"...

To my committee members: Prof. Keogh for showing me that research and science should never be second to administrative trivia, for keeping things in perspective, and always seeing the full scope of life and people beyond the neat little pigeonholes programs try to force them into; Prof. Shelton for his brilliant dedication to science and yet never letting that get in the way of showing kindness and understanding, especially to me in a personal time of need; and Prof. Payne for starting me on my research, for being honest and straightforward in all our discussions, and for always believing in me. I am grateful to have had such exceptional influences

To my advisor, Prof. Roy-Chowdhury, who is not only one of the hardest working people I know but who also worked tirelessly and relentlessly to help shape this research and, without whose intelligence, guidance, and structure, this work simply would never have been. And, especially, for always having faith in our work and science. I am grateful and honoured to have had his help and guidance.

To my colleagues: Utkarsh Gaur, not only for the splendid friendship but also the

iv

the many, many discussions that helped inspire so many ideas reflected in this dissertation. And to Bi Song and Ting Yueh Jeng for the many moments shared in the lab.

To my beautiful sisters (and their amazing husbands and children), for their love and support through all these years. To my lovely mother, who supported and nurtured me endlessly in this, and every other, endeavour my whole life and helped make me who I am today. To my wonderful father, who is not only a thorough gentleman and the sweetest man I know, but also my inspiration and my idol. And, lastly but mostly, to my gorgeous wife, who brings nothing but joy, love, and dedication to every moment of my life, without whom I would not be here today. She is my lifeline and my tireless support in every thing I dare to dream or try.

I thank all of you, without whose help, I would not have been here.

To my wife and parents.

ABSTRACT OF THE DISSERTATION

A Physics-Based, Neurobiologically-Inspired Stochastic Framework for Activity
Recognition

by

Ricky Jaineet Sethi

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, August 2009
Professor Amit K. Roy-Chowdhury, Chairperson

We present a multi-disciplinary framework for motion modeling and recognition in machine vision. Building upon the neurobiological model of motion recognition, we propose computational equivalents for the Motion Energy and Form Pathways. We derive the Hamiltonian Energy Signature (HES) from first-principles in physics as the basis for the Motion Energy Pathway. The Form Pathway is modeled using existing low-level feature descriptors based on shape, appearance, and gradients. We propose an Integration methodology to combine both pathways using a variant of the Feature Integration and Biased Competition neurobiological models, which we implement via statistical hypothesis testing using the bootstrap. We also show the extensibility of our physics-based approach by proposing a physically-significant, compact representation for the gait of a person called the Gait Action Image (GAI), which is based on core physics principles employed in the HES formulation. We then show the generalizability of our neurobiologically-inspired integration framework by casting the GAI within this infrastructure.

Since Motion and image analysis are both important for activity recognition in

video, we also develop a new approach that extends the Hamiltonian Monte Carlo (HMC) to allow us to simultaneously search over the combined motion and image space in a concerted manner using well-known Markov Chain Monte Carlo (MCMC) techniques. For motion analysis in video, we use tracks generated from the video to calculate the Hamiltonian equations of motion for the systems under study, thus utilizing analytical Hamiltonian dynamics to derive a physically significant HMC algorithm which can be used for activity analysis. We then use image analysis to help explore both the motion energy space and the image space by integrating the Hamiltonian energy-based approach with an image-based data-driven proposal to drive the HMC, thereby yielding a Data Driven HMC (DDHMC). We reduce the enormity of the search space by driving the Hamiltonian dynamics-based MCMC with image data in this DDHMC. We also develop the reverse algorithm, which uses motion energy proposals to search the image space. While HMC has been used in other contexts, this is possibly the first work that shows how it can be used for activity recognition in video, taking into account the image analysis results and using the physical motion information of the system. In addition, the DDHMC framework has potential application to other domains where statistical sampling techniques are useful, as we outline in the section on future work.

Experimental validation of the theory is provided on the well-known KTH, Weizmann, and USF Gait datasets with very promising results.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem and Motivation

Activity recognition is difficult because the dynamic nature of video makes it necessary to combine both the image and motion elements represented in videos in order to recognize activities. Integrating these two disparate forms of analysis is a significant problem which we address in this work. Our governing theme in building a solution was to always start from first principles. Following this line of reasoning, we started by examining how one of the most successful systems for activity recognition, the brain, interprets and understands motion.

Understanding activities, indeed, is intuitive for humans. From birth, we observe physical motion in the world around us and create perceptual models to make sense of it. Neurobiologically, we invent a framework within which we understand and interpret human activities [34]. Recent work in neurobiology suggests that the brain also conducts a similar analysis: it examines both the form aspects of motion (e.g., shape, colour, orientation,
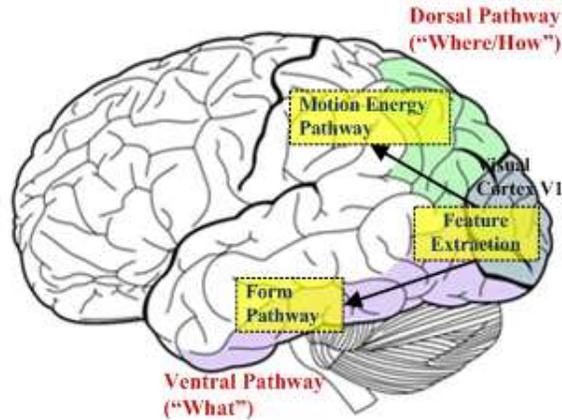
Figure 1.1: Feature extraction in V1 and then division along Motion Energy Pathway (Dorsal) and Form/Shape Pathway (Ventral)

etc.) as well as the motion energy (the kinematics and dynamics) when it attempts motion recognition, as shown in Figure 1.1.

Most modern approaches in activity recognition similarly involve some form of image analysis as well as motion analysis. Image analysis of the form (based on shape, colour, orientation, etc.) is a well-known area of activity recognition. But representing the dynamic element, the motion energy which the brain somehow interprets, is a difficult problem that has not been studied much in machine vision. This dynamic motion element has, however, been widely studied in physics.

Motion, in fact, underlies all activities; human activities are defined by motion. The rigorous study of motion has been the cornerstone of physics for the last 450 years, over which physicists have unlocked a deep, underlying structure of motion. We therefore employ ideas grounded firmly in fundamental physics that are true for the motion of the underlying physical systems considered in video.

Using this physics-based methodology, we propose the computation of **Hamilto-**

nian Energy Signatures (HES) for the various objects (either an entire object or its various parts) involved in an activity, thus representing the motion of each object over the course of an activity as a multi-dimensional time-series. In addition, we develop a new distance metric, called the **S Metric**, which also characterizes the global motion of the object, or the entire scene, with a single, scalar, global value (that can be represented as a series of values if the total video is broken up into shorter time-segments since the S Metric is additive, as shown in Section 2.3); this is exactly what we use in our development of the **Gait Action Image (GAI)**, a spatio-temporal gait representation in which we create an average silhouette image that assigns an intensity value to each point on a person's contour. Both the HES curves and the S-Metric provide a gist of the activity.

This HES is also what we use to represent the Motion Energy pathway of the brain. In addition, we use well-known approaches in activity recognition [2, 66] to represent the Form pathway. Finally, we posit statistical computational models for the Integration of the two pathways (integration of the pathways is still an open question in the neurobiological community [23, 22]) and analyze the relationship between our computational models and some existing neurobiological models.

In order to devleop our framework, we build liberally upon theoretical thrusts from several different disciplines, including Analytical Hamiltonian Mechanics, Neuromorphic Computing and Neurobiology, and, of course, image analysis. The models developed for robotics in [47] provide the basic NMC architecture but are used more for image recognition and analysis. Similarly, Energy-Based Models (EBMs) [39] capture dependencies between variables for *image recognition* by associating a scalar "energy" to each configuration of the variables. Still others [9] take local and global optical flow approaches and compute confi-

dence measures. Also, our physics model goes well beyond an examination of Eulerian or Lagrangian fluid flows, which only yield local (vector) rather than global (integral) properties. Researchers have proposed computational frameworks for integration, e.g., [65], but they have also been restricted to the analysis of single images. The use of DDMCMC shown in [70] would make for an interesting integration approach; motivated by the DDMCMC framework, we propose a new integration approach that takes into account the dynamics of motion. We call this the Data Driven Hamiltonian Monte Carlo (DDHMC).

In terms of human activity recognition [30, 66], some of the cutting edge research uses the fusion of multiple features. In fact, many methods in activity recognition [29, 41, 68, 26, 53, 67] use a combination of motion and form and we are also motivated by the same. Our methodology works in both high- and low-resolution and for densely- and sparsely-distributed objects since all it requires is the (x,y,t) tracks for the various objects. And, unlike methods that compute a low-level signature with dyadic pyramids and Fourier energy [47], our method, as explained in the Appendix and in Section 3.1, computes a physically-significant energy.

In terms of neurobiology, we build liberally upon some of the latest models for integrating the two pathways, both in object recognition and in motion recognition. Neuromorphic computing approaches [47, 58] have examined different integration methodologies in the context of object recognition, including simple pointwise multiplication, as well as exploring more standard neurobiological integration mechanisms such as *feature integration* [64], in which simple visual features are analyzed pre-attentively and in parallel, and *biased competition* [15, 52], which proposes that features compete for attention (at a neuronal level). We also build upon the latest suggestions of integration which propose several alternative

approaches for this [14, 35, 69] in developing our computational variants of integration.

1.2 Contributions

In this work, we make several novel contributions to address the problem of integration of motion and form in order to do activity recognition in video. We develop a computational equivalent for the motion energy pathway of the neurobiological model based upon a fundamental physics formulation. Using the rigorous Hamiltonian framework, we propose Hamiltonian Energy Signatures as an abstract feature for detection of motion energy and activity recognition. In addition, we extend our physical development to create a new spatio-temporal gait representation, called the Gait Action Image. We then create various statistical Integration mechanisms to combine both the motion and form pathways of the neural model. Finally, we develop yet another Integration variant motivated by the DDMCMC but that builds upon a physically-significant Hamiltonian Monte Carlo, which we call the DDHMC. The following are thus the main contributions of this work.

1.2.1 Hamiltonian Energy Signatures for Motion Analysis in Video

We propose the use of the Hamiltonian Energy Signature (HES) based upon the established Hamiltonian framework. Since the Hamiltonian formulation is an abstract representation of the underlying motion, it operates at a higher level than approaches like [31], which build the framework for the basic detection of motion energy; however, the higher, more abstract Hamiltonian representation automatically builds upon the lower levels since it extracts these abstract features directly from the basic tracks of the objects.

While some approaches have proposed the application of the object recognition

5

framework to the analysis of motion by "empirically searching for a suitable representation" [31], our framework, unlike [31, 59, 54], builds upon the rigorous Hamiltonian framework and is applicable in scenarios with and without training data. In addition, in the hierarchical model of the motion energy pathway [22, 21], the first level consists of local motion energy detectors, while the second level analyzes the local structure of the optic flow patterns generated by biological movements, and, lastly, the postulated level of optic flow pattern neurons which are summed in the next, and final, level by the motion pattern neurons. Analogously, our approach can be extended to do more complex modeling of the motion energy pathway, including the higher levels of the hierarchical model [21], using Hamiltion's Action [1] as the HES, as discussed in Section 2.2.2, as well as formulating probabilistic models for Hamilton's Action/HES to make it more robust. Thus, our proposed methodol of HES can be integrated into the hierarchical model along with [31], in order to model the higher, more abstract, levels.

1.2.2   Neurobiologically-Inspired Statistical Integration Strategies

We also propose a complex model for the integration of the two pathways based upon object-recognition models from neurobiology and neuromorphic computing which we represent using three different computational processes for this integration: Total Integration (TI), Partial Integration (PI), and Weighted Integration (WI). One of our proposed mechanisms, WI, does no worse than either of the two pathways individually. To the best of our knowledge, this is the first ever method that proposes a computational equivalent to the integration of the neurobiological models of motion recognition and which is applied to real

---

[1]To avoid confusion, we will use the term Hamilton's Action to refer to S, see equation (2.1), and Action without any prefix to refer to Action as it is normally understood in Action Recognition.

world data. Although approaches like [31], in particular, have helped confirm the benefits of extending biological models for motion recognition to computation, they do not provide a framework for the fusion or integration of the two pathways and concentrate instead on proposing new motion-sensitive units and tuning functions (C_2 features) for the motion pathway in the original neural model [56]. Our approach, however, extends the neurobiological model in [23, 22, 21] to include an integration framework for motion recognition, and provides a rigorous basis for implementing the motion energy pathway using fundamental physics.

1.2.3 Gait Action Image

We further extend our physics-based approach by generalizing the Motion-Energy Image, Motion-History Image, and Gait Energy Image widely used in gait recognition to a physically-significant Gait Action Image (GAI). The GAI is a physics-drive compact representation of the gait of a person and we show the generalizability of our neurobiologically-inspired framework by casting the GAI within this infrastructure.

1.2.4 Data Driven Hamiltonian Monte Carlo Framework

We propose a physically driven method to address the problem of activity recognition by combining the usual energy-based Hamiltonian approach of a Traditional HMC with data-driven proposals derived from video observations. The resulting approach is what we call the Data Driven HMC (DDHMC). DDHMC is a potentially general approach but we mainly demonstrate it in the context of activity recognition in the current work, where we propose to use the DDHMC, on top of video feature tracks (e.g., points on the human sil-

7

houette contour or on body parts in high-resolution or centroids in low-resolution), in order to classify human activities. These tracks may come from a user query or from averaging the Hamiltonian over all the training tracks.

1.2.5   DDHMC Variants

We propose two different forms of the DDHMC: $DDHMC_{Motion}$, which uses motion-based data-driven proposals, and $DDHMC_{Shape}$, which uses shape-based data-driven proposals. Both variants of the DDHMC rely upon a data-driven component to make more informed proposals than the blind proposals generated within a Traditional HMC. These informed proposals, based on the likelihood of a particular track under a Kernel Density or Gibbs estimator, are then used as the data-driven portion of the HMC.

Thus, the initial classification of the activities based on the data-driven portion becomes the first step in a two-step, hierarchical classification scheme implemented by the DDHMC: the data-driven portion does a gross classification and then the Traditional HMC framework does a higher resolution classification with greater granularity. In $DDHMC_{Motion}$, we use a physically-significant Hamiltonian, derived from the tracks, to get the similarity distribution that will help guide the Traditional HMC framework. For $DDHMC_{Shape}$, we use a shape-based similarity distribution to help guide the physically-significant Hamiltonians derived from the tracks that are used within the Traditional HMC framework.

We thus develop the DDHMC framework, as well as its two variants that take into account the image analysis results and the physical motion information of the system, and then apply these models to real world data using challenging datasets.

8

1.3   Organization of this work

The rest of this work is organized as follows: we first cover the details of the Hamiltonian Energy Signature. Then, we examine our proposed Statistical Integration Mechanisms for the Neurobiologically-Inspired Model of Motion. Finally, we detail our development of the Data Driven Hamiltonian Monte Carlo (DDHMC) and describe its variants using the motion and the form separately as proposals. Details of the algorithms, along with evaluation results are provided, followed by future directions for research in the conclusion.

Chapter 2

Hamiltonian Energy Signature

2.1   Introduction

In order to develop a computational framework for the motion energy pathway, we start with a rigorous study of motion using ideas grounded firmly in fundamental physics. We utilize Hamiltonian Dynamics, which is an elegant and powerful alternative formulation of classical mechanics that not only gives the equations of motion for a system but, more importantly, provides greater, and often more abstract, insight about the system. It provides an abstract framework based upon the Principle of Least Action (please see Section 2.2) that can be extended to all other laws of physics (in fact, all fundamental laws of physics can be expressed in terms of a least action principle). Hamilton's equations, using the Hamiltonian H (as shown in Section 2.2), are primarily of interest in establishing basic theoretical results.

Starting from these first principles, we develop a method to extract an abstract representation of the motion of the underlying physical systems we consider in video. Using this physics-based methodology, we propose to use the **Hamiltonian Energy Signatures**

(**HES**) for various objects (either entire objects or the parts of a single object) involved in an activity, thus representing the motion of each object over the course of the activity as a time series.

For example, if we track a person in video, we can compute these HES curves for the centroid of the person (considering the person as an entire object) or consider all the points on the contour of that person's silhouette, thus leading to a multi-dimensional time series (which can, for example, represent the gait of a person). Note that these HES curves can be computed in either the image plane, yielding the Image HES as used in this work, or in the 3D world, giving the Physical HES, depending on the application domain and the nature of the tracks extracted. In either case, the rigorous Hamiltonian framework gives a highly abstract representation for a system and can yield the energy of the system under consideration (please see Appendix for details of the specific conditions). HES curves can therefore model a physically-significant energy that only depends on the image velocities, as conditioned by the neurobiological model [55, 60, 22].

The original neurobiological motion energy formulation of [1], in fact, calculates a "velocity estimate" to first approximation using only the apparent motion in the image plane; this estimate is then "combined with motion information from other channels" in determining the final neurobiological motion percept of subsequent approximations. Our dynamic models also rely only on the image plane and thus approximate the "velocity estimate" to first approximation for mass, potential energy, etc.; subsequent approximations could enhance this estimate with the higher neural pathways of the hierarchical motion energy model [21], which we can model using Hamilton's Action, Lagrangian, and phase space trajectories directly (please see Section 2.2 and the Appendix for details). These HES curves thus

provide the motion energy time series of each object under consideration; as shown in [22], this temporal ordering is essential for sequence selectivity, which is the temporal order of the stimulus sequence, in the neural model of motion recognition.

We also prove that the HES is invariant under an affine transformation. This allows us to use the HES to *categorize* different activities across different domains (high resolution, low resolution, etc.) in a moderately view-invariant manner, as explained below, without requiring separate heuristics (features or representations) for each. In addition, the translation- and scale-invariance, proven for the HES, is another essential component of the neural model of motion recognition [23, 31, 21, 22].

2.2 Hamiltonian Energy Signatures (HES)

One of the most fundamental ideas in theoretical physics is the *Principle of Stationary Action*, also known variously as Principle of Least Action as well as Hamilton's Variational Principle [38]. This is a variational principle that can be used to obtain the equations of motion of a system and is the very basis for most branches of physics, from Analytical Mechanics to Statistical Mechanics to Quantum Field Theory. The basic concept of this principle is to apply the idea of a function whose value remains constant along any path in the configuration space of the system (unless the final and initial points are varied) to Newtonian Mechanics to derive Lagrange's Equations, the equations of motion for the system being studied.

2.2.1 Derivation of HES

Following Hamilton's approach, we define **Hamilton's Action**, S, for motion along a worldline between two fixed physical events (not events in activity recognition) as:

$$S \equiv \int_{t_1}^{t_2} L(q(t), \dot{q}(t), t) dt, \tag{2.1}$$

with q, the generalized coordinates [1], and L, in this case, the **Lagrangian** which, for a conservative system (i.e., energy is conserved), is defined as:

$$L = T - U, \tag{2.2}$$

where, T is the *Kinetic Energy* and U is the *Potential Energy*. The *Hamiltonian function*, derived from **Hamilton's Variational Principle** (please see the Appendix for details), is usually stated most compactly, in generalized coordinates, as [25]:

$$H(q, p, t) = \sum_i p_i \dot{q}_i - L(q, \dot{q}, t), \tag{2.3}$$

where H is the Hamiltonian, p is the generalized momentum, and $\dot{q}$ is the time derivative of the generalized coordinates, q (proof is provided in the Appendix). If the transformation between the Cartesian and generalized coordinates is time-independent, then the Hamiltonian function also represents the total mechanical energy of the system:

$$H(q(t), p(t)) = T(p(t)) + U(q(t)). \tag{2.4}$$

---

[1] Generalized coordinates are the configurational parameters of a system; the natural, minimal, complete set of parameters by which you can completely specify the configuration of the system.

In general, we compute (2.3), which does depend on time, but we can make the assumption (2.4) as a first approximation, as discussed in the Appendix. In this first approximation, the system can be idealized as a holonomic, conservative system (i.e., a system whose state does not depend on the path taken to achieve it; e.g., carrying a box in a gravitational field), unless we deal with velocity-dependent or time-varying potentials.[2]

2.2.2   Application to Motion Recognition

This Hamiltonian is what we utilize as the **Hamiltonian Energy Signature (HES)** to represent the Motion Energy Pathway of a system or sub-system observed in the given video. We end up with a quantity that provides a global description of the motion: the HES (2.3), which can give a simple, intuitive expression for physically significant energy of the motion of each object (i.e., the characteristic time-series curves for each object).

Hamilton's equations, using the Hamiltonian H (as shown in Section 2.2), are primarily of interest in establishing basic theoretical results. We therefore create an *abstract* representation in terms of H, rather than just velocities or trajectories, to see if we can reduce the complexity or make interesting predictions we couldn't with the simpler approach. In fact, we achieve dimensionality reduction (e.g., if we just looked at velocities, N objects will have N tracks), the ability to model complex situations and interactions, and to easily see complex relationships that aren't apparent with just (x,y) trajectories. In fact, by relying on the abstract Hamiltonian framework rather than just the kinetic energy and potential energy, we utilize the greatest advantage of this abstract representation by allowing for theoretical extensions; this is the main reason the Hamiltonian/Least Action formulation lies at the core

---

[2]In fact, even when we cannot make those idealizations (e.g., viscous flows), we can define "generalized potentials" [25] and retain the standard Lagrangian, as in (2.2).

of most modern physics formulations: not because it allows for simpler computations than the Lagrangian/Newtonian framework alone but, rather, because it allows for theoretical extenstions (Section 2.2). In fact, we see an immediate benefit of the theoretical extensions allowed by this abstract represention in the form of the proofs (e.g., view invariance proof) we show in Section 2.3.

Thus, our approach is to segment the video into systems and sub-systems (e.g., whole body of a person, or parts of the body) and, for each of those, get their tracks, from which we compute T and U, and use that to get the HES curve signature, which can then be evaluated further in phase space and the results analyzed accordingly, as shown in Figure 2.1. The phase–space of a system consists of all possible values of the generalized coordinate variables q_i and the generalized momenta variables p_i. If the Hamiltonian is time-independent, then phase space is 2-dimensional, (q,p); if the Hamiltonian is time-dependent, then phase space is 3-dimensional, (q,p,t) [62].

We use the video to gain knowledge of the physics and use the physics to capture the Motion Energy Pathway of the object being observed via the HES. In order to compute the HES, we use the tracks from the video to compute the kinematic quantities that drop out of the Lagrangian formalism, thus giving a theoretical basis for examination of their energy from (x,y,t).

Figure 2.1: Tracks to Hamiltonian to Phase Space: the phase space of a system consists of all possible values of the coordinates, which can be (q,p) or (q,p,t), for example; we may also look at modified phase plots of (H,t), (H,q,p), etc.

15

2.2.3 Examples

For example, in the general case when U $\neq$ 0, the Lagrangian, T - U, of a single particle or object acting under a constant force, F (e.g., for a gravitational field, g, F=mg) over a distance, x, is:

$$L(x(t), \dot{x}(t)) = \tfrac{1}{2}mv^2 - Fx,$$
$$\text{with } x = x_o + v_o t + \tfrac{1}{2}at^2 \text{ and } a = \tfrac{F}{m}. \tag{2.5}$$

We now use this Lagrangian to calculate Hamilton's Action for the general system:

$$S = \int_{t_a}^{t_b} L dt = \int_{t_a}^{t_b} \left(\tfrac{1}{2}m(v_o^2 + 2v_o \tfrac{F}{m}t) - F(x_o + v_o t) \right) dt$$
$$= \tfrac{1}{2}mv_o^2(t_b - t_a) - Fx_o(t_b - t_a) \tag{2.6}$$

Using Hamilton's Variational Principle on (2.6) for a gravitational force yields (with y being the vertical position, which is determined from the tracks):

$$H = T + U = \frac{1}{2}mv_o^2 + mgh = \frac{1}{2}mv_o^2 + mg(y_b - y_a) \tag{2.7}$$

Using the Hamiltonian equations of motion for various objects (either entire objects or the parts of a single object) involved in activities observed in video, we can easily visualize these HES curves for simple cases, like the box exchange activity in Figure 2.2(a) between two people. Plots of the HES curves can give us a sense of the energies associated with this activity, both in the idealized case (Figure 2.2(b)) and for the experimentally observed case (Figure 2.2(c)). Here, as a first approximation, we treat m as a scale factor and set it to unity; in future, we can estimate mass using the shape of the object or other heuristics, including estimating it as a Bayesian parameter. In addition, mass is not as significant when

we consider the same class of objects, since it factors out during the process of comparing two sequences.

For more complex interactions, we can even use any standard, conservative/non-conservative model for U (e.g., as a spring with $U = \frac{1}{2}kx^2$ for elastic interactions) whereas, for the simplest case of a free particle, we are left with just the T; this is equivalent to situations with videos of activities happening in far-field where we can only discern the motion of objects on a ground plane. Even for these cases, we can also plot Hamilton's Action vs. time as the HES curve since the partial derivative of the Action is energy [38]. For example, we can see an example of two cars, one following the other, in Figure 2.3. Here, the first car, whose trajectory is labeled in orange, is the lead car and executes a U-turn; the second car, trajectory in blue, follows it and also makes a U-turn, whereas the third car, whose trajectory is in red, follows it for a while and then turns away.

On the other extreme, we can compute more complex interactions between the points on a person's contour/body joints if we are studying gait or the kinematics of the different body parts. The HES curve generated for each person is actually a multi-dimensional vector composed of HES curves for the different points on the person, as shown in Figure 2.4.

The HES formulation allows us to analyze activities based purely on their motion. While it will, in general, not be sufficient for complete activity recognition, it formalizes a first level of discrimination using only the motion information and provides a framework for theoretical extensions. We thus derive the HES curves for all given objects in video and we can compare their characteristic HES curves using a Dynamic Time Warping (DTW) algorithm.
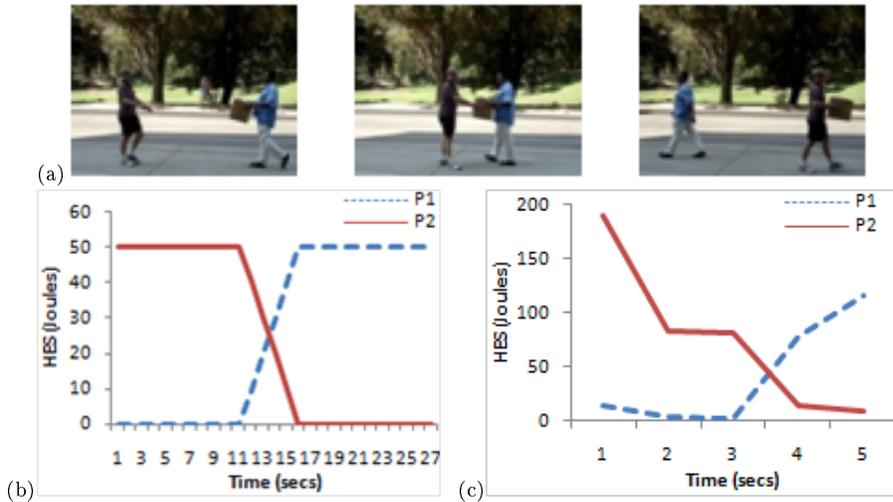
Figure 2.2: (a) Box Exchange experiment video: (b) Ideal vs (c) Actual Hamiltonian curves. Here we see two people exchanging a box in (a). Plots of the Hamiltonian equations of motion can give us a sense of the energies associated with this activity, both in the idealized case (b) and for the experimentally observed case (c).

2.3   Theoretical Derivation of HES and its Properties

In this section, we build upon the derivation of the HES and show some of its properties in detail. We start off by detailing the Hamilton's Variational Principle, based upon the Principle of Least Action, which we use to derive the Hamiltonian and that forms the basis of the HES (shown in the Appendix). Next, we prove the view invariance of the Action/Lagrangian framework, which we demonstrate experimentally using the KTH dataset in Section 2.4.

2.3.1   View Invariance of the Action

In this section, we show the invariance of the Action which, by Equations (A.1)-(A.4), applies to H and thus, by extension, to HES, as well. We start off by using the invariance properties of the Lagrange equations; in particular, one of the properties of the
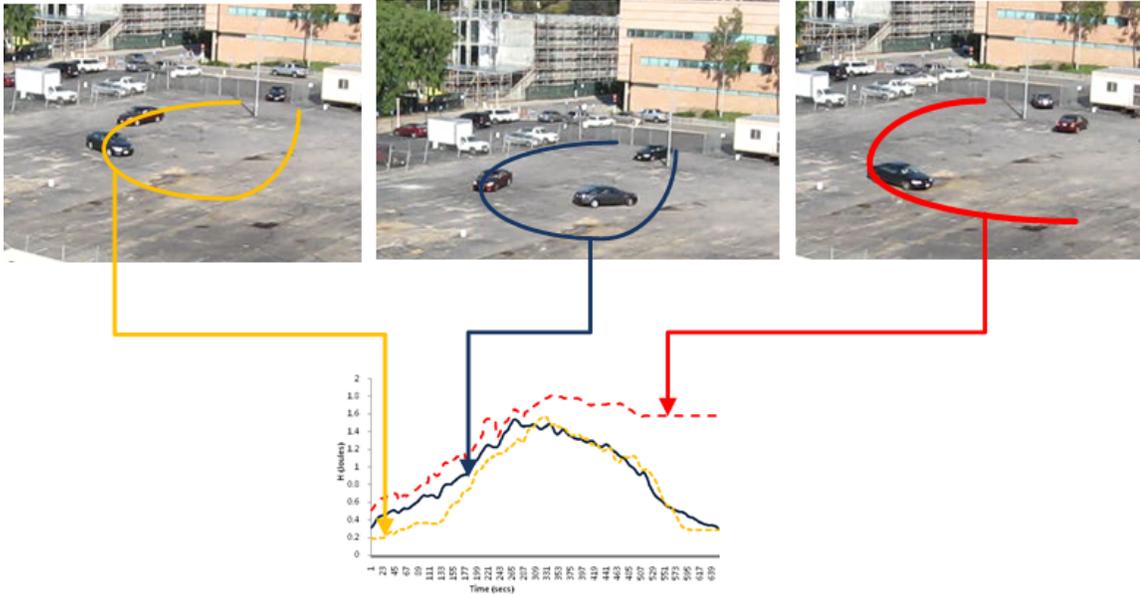
Figure 2.3: Two cars following: the first car, whose trajectory is labeled in orange, is the lead car and executes a U-turn; the second car, trajectory in blue, follows it and also makes a U-turn, whereas the third car, whose trajectory is in red, follows it for a while and then turns away.

Lagrange Equations is their form-invariance under coordinate transformations, especially under translations and rotations. This, in turn, implies the Action is also invariant under a **Euclidean Transform** (translation and rotation). We can also see this invariance from first principles by starting with the three fundamental facts of the physical world:

1. Homogeneity of Time: any instant of time is equivalent to another

2. Homogeneity of Space: any location in space is equivalent to another

3. Isotropy of Space: any direction in space is equivalent to another

Two consequences follow from these three facts, for which there is no evidence of the contrary:

1. The mechanical properties of a closed system are unchanged by any parallel displace-

19
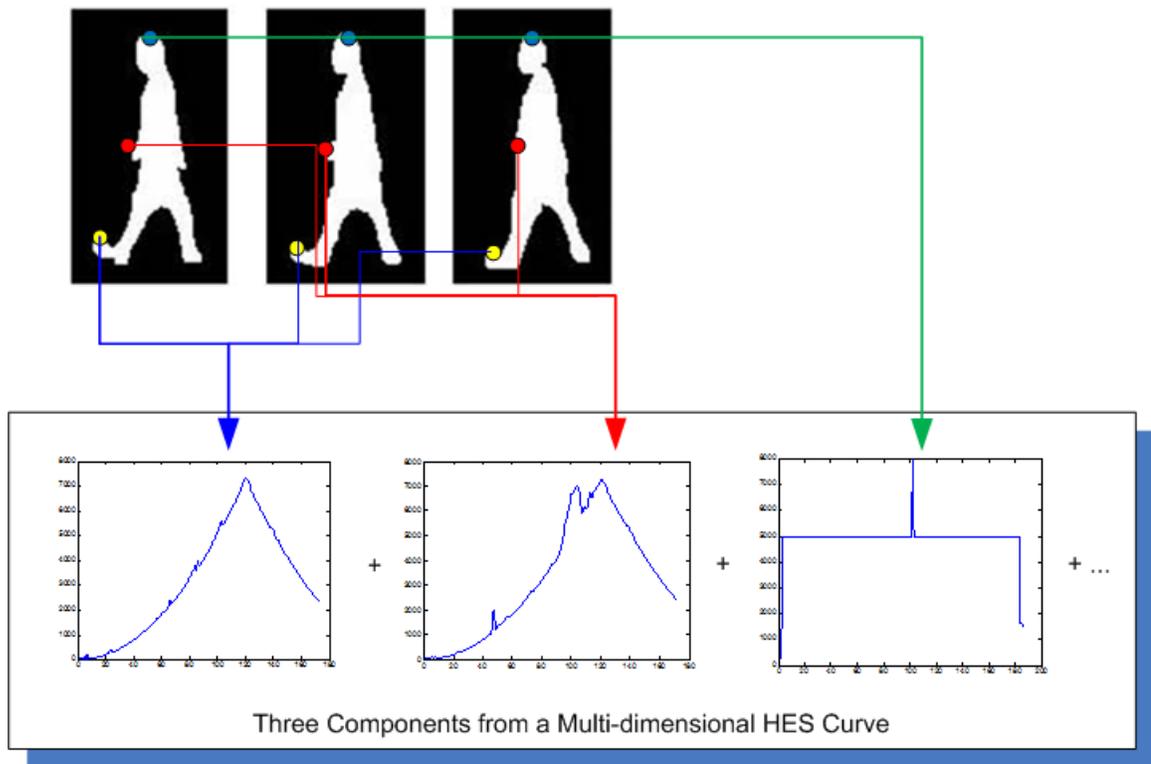
Figure 2.4: Vector of HES curves for a person's silhouette

ment of the entire system in space.

2. The mechanical properties of a closed system do not vary when it is rotated as a whole in any manner in space.

And so, three properties of the Lagrangian follow:

1. The Lagrangian of a closed system does not depend explicitly on time.

2. The Lagrangian of a closed system remains unchanged under translations.

3. The Lagrangian of a closed system remains unchanged under rotations.

We use these basic principles in the following approach. For Lagrangian Invariance under Special Affine Transformations (translation and rotation), let the solution of the Lagrange Equations in the original coordinate system be:

$$x = x(t), v = v(t) \Rightarrow S = \int_{t_1}^{t_2} L(x(t), v(t)) dt \tag{2.8}$$

The solution of the Lagrange Equations for a displaced system therefore is:

$$\tilde{x} = x_0 + Rx(t), \tilde{v} = Rv(t) \tag{2.9}$$

The Action calculated on the solution of the displaced system is:

$$\tilde{S} = \int_{t_1}^{t_2} L(\tilde{x}, \tilde{v}) dt = \int_{t_1}^{t_2} L(x_0 + Rx(t), Rv(t)) dt \tag{2.10}$$

Invariance of the Lagrangian under translation gives:

$$\tilde{S} = \int_{t_1}^{t_2} L(Rx(t), Rv(t)) dt \tag{2.11}$$

21

Invariance of the Lagrangian under rotation gives:

$$\tilde{S} = \int_{t_1}^{t_2} L(x(t), v(t)) dt = S \tag{2.12}$$

Since S is the Action calculated on the solution in the original system of coordinates, this shows the invariance of the Action under rotational and translational transformations:

$$\tilde{S} = S \tag{2.13}$$

This also applies, by Equations (A.1)-(A.4), to H and hence shows that the HES computed from 3D points is invariant to rigid translational and rotational transformations. The HES computed from image parameters is invariant to 2D translations, rotations, and skew on the image plane as these properties are further proven for the Lagrangian in Section 2.3.1.1. We thus show that the 3D Hamiltonian is invariant to rigid 3D transformations and the Image Hamiltonian is invariant to 2D transformations.

2.3.1.1   Affine Invariance of the Lagrangian and the Action

The previous section depends on the rotational and translational invariance of the Lagrangian and so, here we show that the Lagrangian is invariant under an arbitrary affine transform of World Coordinates (e.g., any combination of scaling, rotation, transform, and/or shear). This also applies to the Hamiltonian by the Legendre Transform of (A.4) since, as shown in Section A, the Legendre Transform is used in classical mechanics to derive the Hamiltonian formulation from the Lagrangian formulation and vice versa. Thus, we have the Lagrangian:

$$L = T - U \tag{2.14}$$

with Kinetic Energy, T, and Potential Energy, U. The Lagrangian is a function:

$$L = L\left(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3\right) \tag{2.15}$$

where x_1 x_2 x_3 are the world coordinates and $\dot{x}_i = \frac{dx_i}{dt}$ are the time derivatives. We first do an affine transform:

$$y_i = \sum_j C_{ij} x_j + d_i \tag{2.16}$$

Because this is an affine transform, the determinant of the matrix C_{ij} is non-zero; i.e., $|C| \neq 0$. Then, the inverse matrix exists:

$$A = C^{-1} \tag{2.17}$$

Now we can get the formulae for x_i :

$$x_i = \sum_j A_{ij} y_j + b_i \tag{2.18}$$

The new Lagrangian is a complex function:

$$L'\left(y_1, y_2, y_3, \dot{y}_1, \dot{y}_2, \dot{y}_3\right) = L\left(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3\right) \tag{2.19}$$

Where $x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3$ are functions of $y_1, y_2, y_3, \dot{y}_1, \dot{y}_2, \dot{y}_3$:

$$x_1 = x_1\left(y_1, y_2, y_3\right) \tag{2.20}$$

$$x_2 = x_2 \left(y_1, y_2, y_3\right) \tag{2.21}$$

$$x_3 = x_3 \left(y_1, y_2, y_3\right) \tag{2.22}$$

$$\dot{x}_1 = \dot{x}_1 \left(\dot{y}_1, \dot{y}_2, \dot{y}_3\right) \tag{2.23}$$

$$\dot{x}_2 = \dot{x}_2 \left(\dot{y}_1, \dot{y}_2, \dot{y}_3\right) \tag{2.24}$$

$$\dot{x}_3 = \dot{x}_3 \left(\dot{y}_1, \dot{y}_2, \dot{y}_3\right) \tag{2.25}$$

As we can see, the Lagrangian stays invariant: $L = L'$ as it is just a substitution of variables and it does not change the Lagrangian value.

Now we prove that Euler-Lagrange equation and Action principle are invariant under an affine transform. In Lagrangian mechanics, the basic principle is not Newton's equation but the Action principle: the Action is a minimal, as per (A.1). The Action principle then leads to the Euler-Lagrange equation:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i} = \frac{\partial L}{\partial x_i} \tag{2.26}$$

where $i = 1, 2, 3$, x_1 x_2 x_3 are the world coordinates, and $\dot{x}_i = \frac{dx_i}{dt}$ are the time derivatives.

We now show that the Action principle, and then Euler-Lagrange equations, are invariant under an affine transform. We have an affine transform, as in (2.16): $y_i =$

$\sum_j C_{ij}x_j + d_i$. Again, because this is an affine transform, the determinant of the matrix C_{ij} is non-zero $|C| \neq 0$. Then, the inverse matrix exists, once again: $A = C^{-1}$ and we can get the formulae for x_i: $x_i = \sum_j A_{ij}y_j + b_i$. The new Lagrangian is again a complex function:

$$L'\left(y_1, y_2, y_3, \dot{y}_1, \dot{y}_2, \dot{y}_3\right) = L\left(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3\right) \qquad (2.27)$$

Where $x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3$ are functions of $y_1, y_2, y_3, \dot{y}_1, \dot{y}_2, \dot{y}_3$ as in (2.20)- (2.25) with time derivatives:

$$\dot{x}_i = \sum_j A_{ij}\dot{y}_j \qquad (2.28)$$

Note:

$$\frac{\partial x_i}{\partial y_j} = A_{ij} \qquad (2.29)$$

$$\frac{\partial \dot{x}_i}{\partial \dot{y}_j} = A_{ij} \qquad (2.30)$$

We now multiply equation (2.26) by A_{ij} and sum over i:

$$\sum_i \frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i}A_{ij} = \sum_i \frac{\partial L}{\partial x_i}A_{ij} \qquad (2.31)$$

$$\sum_i \frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i}\frac{\partial \dot{x}_i}{\partial \dot{y}_j} = \sum_i \frac{\partial L}{\partial x_i}\frac{\partial x_i}{\partial y_j} \qquad (2.32)$$

Using the rule of derivatives of complex functions:

25

$$\sum_i \frac{\partial L}{\partial x_i} \frac{\partial x_i}{\partial y_j} = \frac{\partial L'}{\partial y_j} \qquad (2.33)$$

$$\sum_i \frac{\partial L}{\partial \dot{x}_i} \frac{\partial \dot{x}_i}{\partial \dot{y}_j} = \frac{\partial L'}{\partial \dot{y}_j} \qquad (2.34)$$

Then

$$\frac{d}{dt} \frac{\partial L'}{\partial y_j} = \frac{\partial L'}{\partial \dot{y}_j} \qquad (2.35)$$

which is the same equation as (2.26), hence it is invariant under an affine transform and thus the Action principle is also invariant under an affine transform. Thus, Lagrangian Invariance implies Action Invariance which implies the invariance of the Euler-Lagrange equations which, in turn, implies the invariance of the HES.

2.4   View Invariance Experiments

In Section 2.3, we proved the invariance of the Hamiltonian under an affine transformation. Since the 3D Hamiltonian is invariant to an affine transformation, this implies the 2D image Hamiltonian is moderately view-invariant, as long as the 3D transformation does not change too much; i.e., for some small transformations, it will be moderately view-invariant.

In this section, we provide empirical confirmation of the theory by showing experimental results demonstrating the view invariance of the HES using the well-known KTH dataset. The KTH dataset (`http://www.nada.kth.se/cvap/actions/`) contains six types of human actions (walking, jogging, running, boxing, hand waving, and hand clapping) per-

		Walking (A1)				Jogging (A2)				Running (A3)				Boxing (A4)				Clapping (A5)				Handwaving (A6)			
		V1	V2	V3	V4	V1	V2	V3	V4	V1	V2	V3	V4	V1	V2	V3	V4	V1	V2	V3	V4	V1	V2	V3	V4
Walking (A1)	V1	■	0.003	0.002	0.001	0.067	0.112	0.133	0.063	0.595	0.537	0.984	0.976	0.750	0.546	0.884	0.852	1.000	0.980	0.935	0.833	0.457	0.316	0.395	0.382
	V2	0.003	■	0.004	0.004	0.095	0.121	0.142	0.073	0.763	0.691	0.820	0.816	0.780	0.600	0.898	0.870	1.000	0.983	0.943	0.853	0.521	0.396	0.466	0.454
	V3	0.002	0.004	■	0.001	0.061	0.088	0.123	0.043	0.513	0.455	0.803	0.790	0.756	0.559	0.887	0.855	1.000	0.981	0.936	0.837	0.471	0.335	0.411	0.399
	V4	0.001	0.004	0.001	■	0.061	0.100	0.125	0.050	0.549	0.492	0.951	0.941	0.760	0.566	0.889	0.858	1.000	0.981	0.937	0.840	0.480	0.347	0.422	0.410
Jogging (A2)	V1	0.067	0.095	0.061	0.061	■	0.010	0.014	0.004	0.413	0.400	0.544	0.613	0.707	0.472	0.864	0.826	1.000	0.977	0.924	0.804	0.371	0.221	0.310	0.299
	V2	0.112	0.121	0.088	0.100	0.010	■	0.001	0.015	0.354	0.327	0.430	0.526	0.698	0.459	0.860	0.821	1.000	0.976	0.921	0.798	0.357	0.209	0.299	0.283
	V3	0.133	0.142	0.123	0.125	0.014	0.001	■	0.021	0.340	0.313	0.413	0.522	0.694	0.454	0.857	0.818	1.000	0.976	0.920	0.795	0.352	0.209	0.295	0.277
	V4	0.063	0.073	0.043	0.050	0.004	0.015	0.021	■	0.409	0.378	0.579	0.577	0.711	0.480	0.866	0.829	1.000	0.977	0.925	0.807	0.380	0.231	0.318	0.307
Running (A3)	V1	0.595	0.763	0.513	0.549	0.413	0.354	0.340	0.409	■	0.028	0.021	0.074	0.664	0.412	0.843	0.800	1.000	0.973	0.911	0.774	0.307	0.173	0.241	0.224
	V2	0.537	0.691	0.455	0.492	0.400	0.327	0.313	0.378	0.028	■	0.006	0.024	0.670	0.418	0.846	0.804	1.000	0.974	0.913	0.778	0.312	0.170	0.249	0.231
	V3	0.984	0.820	0.803	0.951	0.544	0.430	0.413	0.579	0.021	0.006	■	0.018	0.649	0.393	0.835	0.789	1.000	0.971	0.907	0.763	0.294	0.177	0.222	0.209
	V4	0.976	0.816	0.790	0.941	0.613	0.526	0.522	0.577	0.074	0.024	0.018	■	0.655	0.400	0.839	0.794	1.000	0.972	0.909	0.768	0.294	0.162	0.225	0.209
Boxing (A4)	V1	0.833	1.000	0.846	0.852	0.590	0.541	0.508	0.609	0.390	0.424	0.333	0.366	■	0.014	0.002	0.001	0.008	0.007	0.004	0.000	0.040	0.141	0.070	0.078
	V2	0.785	1.000	0.810	0.822	0.508	0.458	0.427	0.530	0.310	0.339	0.258	0.286	0.012	■	0.033	0.029	0.051	0.048	0.040	0.025	0.000	0.082	0.020	0.027
	V3	0.854	1.000	0.863	0.867	0.628	0.580	0.546	0.646	0.432	0.466	0.373	0.408	0.003	0.027	■	0.000	0.001	0.001	0.000	0.000	0.055	0.163	0.087	0.097
	V4	0.849	1.000	0.860	0.864	0.620	0.572	0.538	0.638	0.423	0.457	0.364	0.399	0.002	0.025	0.000	■	0.002	0.002	0.001	0.000	0.052	0.158	0.083	0.093
Clapping (A5)	V1	0.867	1.000	0.875	0.877	0.653	0.606	0.572	0.670	0.460	0.495	0.401	0.437	0.007	0.035	0.001	0.002	■	0.000	0.000	0.003	0.066	0.175	0.099	0.109
	V2	0.865	1.000	0.873	0.875	0.649	0.602	0.568	0.667	0.456	0.490	0.396	0.433	0.006	0.034	0.001	0.002	0.000	■	0.000	0.002	0.065	0.173	0.097	0.107
	V3	0.860	1.000	0.868	0.871	0.639	0.592	0.558	0.657	0.445	0.479	0.386	0.422	0.004	0.031	0.000	0.001	0.000	0.000	■	0.001	0.060	0.169	0.093	0.103
	V4	0.846	1.000	0.857	0.862	0.615	0.566	0.532	0.633	0.417	0.451	0.359	0.393	0.001	0.024	0.000	0.000	0.003	0.002	0.001	■	0.051	0.156	0.081	0.090
Handwaving (A6)	V1	0.756	1.000	0.785	0.801	0.459	0.409	0.381	0.482	0.265	0.291	0.223	0.241	0.055	0.003	0.089	0.081	0.121	0.115	0.103	0.076	■	0.045	0.000	0.004
	V2	0.680	1.000	0.727	0.755	0.348	0.304	0.286	0.374	0.184	0.196	0.165	0.162	0.262	0.118	0.352	0.331	0.425	0.413	0.385	0.318	0.048	■	0.008	0.000
	V3	0.730	1.000	0.766	0.788	0.431	0.386	0.360	0.454	0.237	0.263	0.193	0.211	0.117	0.035	0.166	0.154	0.211	0.203	0.186	0.147	0.007	0.023	■	0.000
	V4	0.724	1.000	0.763	0.785	0.426	0.375	0.347	0.449	0.226	0.251	0.185	0.201	0.135	0.045	0.189	0.176	0.238	0.230	0.211	0.168	0.012	0.017	0.000	■

Figure 2.5: KTH Distance Matrix where we highlight the lowest relative values in a row. This shows the matching of similar activities despite view changes. Please note this is not necessarily symmetric because we do the analysis row-wise using training and classification.

formed several times by 25 subjects in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes, and indoors. All sequences are taken over homogeneous backgrounds with a static camera with a 25fps frame rate. The sequences are downsampled to a spatial resolution of 160x120 pixels and have a length of four seconds on average. We use these to demonstrate the view invariance of the Motion Energy Pathway and present a distance matrix for all six actions in Figure 2.5.

The Activities are labeled as follows: Walking (A1), Jogging (A2), Running (A3), Boxing (A4), Clapping (A5), and Handwaving (A6). Each activity has four different views, labeled V1 through V4. As can be seen in Figure 2.5, there is significant matching between the same activity from different views, with lower scores indicating greater similarity and the lowest relative scores highlighted in the figure. There are occasional exceptions, especially for activities involving excessive translational motion, as there are sometimes very few frames

(as few as 10-20) with a sample rate of 25fps; i.e., there are too few frames for a completely reliable calculation of the HES curve. Also, for the clapping and boxing activities, the tracker lost tracks because it was difficult to keep consistent tracking when only the hands were moving; thus, the manual tracking also led to significant errors, in addition the issue of too few frames for those sets, as well. However, our model is able to distinguish between different activities, regardless of view, and matches the same activity, again, irrespective of the different view. We thus demonstrate the view invariance of the HES/S-Metric.

2.5   A Special Application: Gait Action Image (GAI)

In this section, we introduce a special extension of the Hamiltonian framework as applied to the problem of gait recognition. Compact, image-based representations of gait have been an area of research, where Motion History Images (MHI), Motion Energy Images (MEI), and Gait Energy Images (GEI) are three popular descriptors [7, 28]. Building upon current approaches that use MHI, MEI, and GEI, as well as the analysis of the dense optical flow by [40], we develop a spatio-temporal gait representation that builds upon all three of these but is based upon the fundamental Action (please see equation (2.1)) of a system since Action is related to motion; we call our new representation the Gait Action Image (GAI). We then go on to prove that Action is a Norm and demonstrate the addivity of Actions, which we use in the development of the GAI and its distance measure. GAI can then also serve as the motion pathway in the neurobiological model of motion.

2.5.1 Derivation of GAI

MHI and MEI were proposed by [7] as formulations for human movement recognition. Both MEI and MHI are vector-valued images where the vector value at each pixel is a function of the motion properties at that particular location in an image sequence. MEI is a binary image which represents where motion has occured in an image sequence:

$$MEI_\tau(x,y,t) = \bigcup_{i=0}^{\tau-1} D(x,y,t-i) \tag{2.36}$$

where $D(x,y,t-i)$ is a binary sequence indicating regions of motion, τ is the length of time, t is a particular moment in time, and (x,y) are the values of the 2D image coordinates. In similar fashion, MHI is a grey-level image which represents how a motion region in the image is moving:

$$MHI_\tau(x,y,t) = \begin{cases} \tau, \text{ if } D(x,y,t) = 1; \\ \\ max\left\{0, MHI_\tau(x,y,t-1) - 1\right\}, \text{ otherwise.} \end{cases} \tag{2.37}$$

Similarly, GEI [28] is a widely used spatio-temporal gait representation that has been shown to be a robust gait descriptor for gait recognition. GEI builds upon the approach of [7], who proposed Motion-Energy Image (MEI) and Motion-History Image (MHI) formulations for human movement recognition. Both MEI and MHI assign a value to each pixel as a function of the motion properties at that location in an image sequence. GEI also creates an average silhouette image that assigns an intensity value to each pixel; it does so by starting with a size-normalized and horizontally-aligned binary silhouette, $B(x,y,t)$, and defines a grey-level GEI, $GEI(x,y)$, as:

$$GEI(x,y) = \frac{1}{N} \sum_{t=1}^{N} B(x,y,t) \tag{2.38}$$

where N is the number of frames in a complete cycle of the sequence, t is the frame number of the sequence, and (x,y) are the 2D image coordinates. Although, in general, MEI and MHI are different motion representations than GEI, a correspondence between the binary version of GEI and a modified MEI can be shown [27].

In this work, we use the ideas behind GEI, MEI, and MHI as motivation to extend our physics-based approach by generalizing them to a physically-significant **Gait Action Image** (GAI). The GEI is an averaged silhouetted summed over the temporal sequence; Hamilton's Action is a similarly integrated quantity over a specific time interval, as shown in (2.1). We combine these two ideas by computing Hamilton's Action for each point on the human silhouette contour or body parts in a given cycle as:

$$GAI(x,y) = GAI(q) = \frac{1}{N} \int_{t=1}^{N} L(q(t), \dot{q}(t), t) dt \tag{2.39}$$

where N is again the number of frames in a complete cycle and q and $\dot{q}$ are the generalized coordinate and generalized velocity, respectively (L is again the Lagrangian). Following the example of [27, 43], we measure the similarity between the gallery (training) and probe (test) templates of two gait sequences, GAI_g and GAI_p respectively, by calculating their distance as the normalized matching error:

$$D(GAI_g, GAI_p) = \frac{\sum_{x,y} |GAI_g(x,y) - GAI_p(x,y)|}{\sqrt{\sum_{x,y} GAI_g(x,y) \sum_{x,y} GAI_p(x,y)}} = \frac{\sum_{q} |GAI_g(q) - GAI_p(q)|}{\sqrt{\sum_{q} GAI_g(q) \sum_{q} GAI_p(q)}} \tag{2.40}$$

where $\sum_{x,y} |GAI_g(x,y) - GAI_p(x,y)|$ is the matching error between two GAIs (sum of the magnitudes of the difference between two GAIs) and $\sum_{x,y} GAI(x,y)$ is the total energy/action in a GAI. Because GAI mirrors the GEI, MEI, and MHI formulations and representations so closely, all the extensions and proposed algorithms for them should be immediately extensible to GAI, as well. In addition, we can use distance or similarity measures computed using GAI directly in our Integration framework by combining that similarity distribution with one of the standard shape/form methodologies, as described in Section 3.2. We show an example of the GAI in Figure 2.6 and show experimental results of using the GAI as the motion pathway and shape sequence for the form pathway in the Integration in Section 3.5.

2.5.2   Proof Action is a Norm

Here we show that the Action is a norm that induces a distance in the vector space of the velocities. This will be an important property for use in the GAI to justify comparing different gait signatures using scalar values, as in (2.40). Given the Action for a free particle (U=0):

Figure 2.6: Examples of the Gait Action Image (GAI) formed by averaging the row of silhouettes, with darker blues representing higher Action values and lighter blues representing lower Action values for points on the contour.

$$S_f = \int_{t_a}^{t_b} L\left(q, \dot{q}, t\right) dt = \frac{1}{2} m \frac{(x_b - x_a)^2}{t_b - t_a}$$

$$= \frac{1}{2} m v^2 \left(t_b - t_a\right) = \frac{1}{2} m v^2 \Delta t \tag{2.41}$$

We want to prove that $\sqrt{S_f(m, v, \Delta t)} = \|v\|_{m,\Delta t}$ is a norm on the vector space $\mathbb{R}^3$ of the velocities. We have to prove that the following holds:

1. $\|v\|_{m,\Delta t} \geq 0, \|v\|_{m,\Delta t} = 0$ if and only if $v = 0$

2. $\|\lambda v\|_{m,\Delta t} = |\lambda| \, \|v\|_{m,\Delta t}$

3. Given two free particles, $(m_1, v_1), (m_2, v_2)$, and the system made up by the two, with TM the total mass and CM the center of mass:

$$(m_{TM}, v_{CM}), \|v_{CM}\|_{m_{TM},\Delta t} \leq \|v_1\|_{m_1,\Delta t} + \|v_2\|_{m_2,\Delta t} \tag{2.42}$$

Properties 1 and 2 are trivially true. In order to prove Property 3, the **triangle inequality**, let $(m_1, v_1), (m_2, v_2)$ be two free particles. The system of the two particles is characterized by its total mass (TM), its center of mass (CM), and the velocity of its center of mass:

$$m_{TM} = m_1 + m_2,$$

$$x_{CM} = \frac{m_1 \vec{x}_1 + m_2 \vec{x}_2}{m_1 + m_2}, \quad v_{CM} = \frac{m_1 \vec{v}_1 + m_2 \vec{v}_2}{m_1 + m_2} \tag{2.43}$$

and the Action of the two particles, from (2.41), considered as one gives the norm:

$$\|v_{CM}\|_{m_{TM},\Delta t} = \sqrt{\frac{m_{TM} \vec{v}_{CM}^2}{2} \Delta t} \tag{2.44}$$

In order to prove the triangle inequality, we need:

32

$$\|v_{CM}\|^2_{m_{TM},\Delta t} = \frac{m_{TM}\vec{v}^2_{CM}}{2}\Delta t = \frac{1}{2}\left(m_1+m_2\right)\left(\frac{m_1\vec{v}_1+m_2\vec{v}_2}{m_1+m_2}\right)^2\Delta t$$

$$= \frac{1}{2}\frac{m_1^2\vec{v}_1^2+2m_1m_2\vec{v}_1\cdot\vec{v}_2+m_2^2\vec{v}_2^2}{m_1+m_2}\Delta t \tag{2.45}$$

and

$$\left(\|v_1\|_{m_1,\Delta t}+\|v_2\|_{m_2,\Delta t}\right)^2 = \frac{m_1\vec{v}_1^2}{2}\Delta t + \sqrt{m_1m_2}v_1v_2\Delta t + \frac{m_2\vec{v}_2^2}{2}\Delta t \tag{2.46}$$

Their difference gives:

$$\left(\|v_1\|_{m_1,\Delta t}+\|v_2\|_{m_2,\Delta t}\right)^2 - \|v_{CM}\|^2_{m_G,\Delta t}$$

$$= \frac{m_1\vec{v}_1^2}{2}\Delta t + \sqrt{m_1m_2}v_1v_2\Delta t + \frac{m_2\vec{v}_2^2}{2}\Delta t - \frac{1}{2}\frac{m_1^2\vec{v}_1^2+2m_1m_2\vec{v}_1\cdot\vec{v}_2+m_2^2\vec{v}_2^2}{m_1+m_2}\Delta t \tag{2.47}$$

$$= \frac{m_1m_2\vec{v}_1^2+m_1m_2\vec{v}_2^2+2(m_1+m_2)\sqrt{m_1m_2}v_1v_2-2m_1m_2\vec{v}_1\cdot\vec{v}_2}{2(m_1+m_2)}\Delta t$$

The difference is positive because:

$$m_1m_2\vec{v}_1^2,\ m_1m_2\vec{v}_2^2 > 0 \tag{2.48}$$

and

$$2(m_1+m_2)\sqrt{m_1m_2}v_1v_2 - 2m_1m_2\vec{v}_1\cdot\vec{v}_2 > 0 \tag{2.49}$$

with strict inequality, because of the *Cauchy-Schwarz inequality*:

$$v_1v_2 \geq \vec{v}_1\cdot\vec{v}_2 \tag{2.50}$$

and because the *arithmetic mean* is greater than the *geometric mean*:

33

$$m_1 + m_2 \geq 2\sqrt{m_1 m_2} > \sqrt{m_1 m_2} \tag{2.51}$$

Substituting these in and expanding, gives:

$$
\begin{aligned}
&2(m_1 + m_2)\sqrt{m_1 m_2}v_1 v_2 - 2m_1 m_2 \vec{v}_1 \cdot \vec{v}_2 \\
&= 2\sqrt{m_1 m_2}((m_1 + m_2)v_1 v_2 - \sqrt{m_1 m_2}\vec{v}_1 \cdot \vec{v}_2) \\
&> 2\sqrt{m_1 m_2}(\sqrt{m_1 m_2}v_1 v_2 - \sqrt{m_1 m_2}\vec{v}_1 \cdot \vec{v}_2) \\
&= 2m_1 m_1(v_1 v_2 - \vec{v}_1 \cdot \vec{v}_2) \geq 0
\end{aligned}
\tag{2.52}
$$

Putting it all together, this finally yields:

$$\left(\|v_1\|_{m_1,\Delta t} + \|v_2\|_{m_2,\Delta t} \right)^2 - \|v_{CM}\|_{m_{TM},\Delta t}^2 > 0 \tag{2.53}$$

And so, we have the triangle inequality:

$$\|v_{CM}\|_{m_{TM},\Delta t} < \|v_1\|_{m_1,\Delta t} + \|v_2\|_{m_2,\Delta t} \tag{2.54}$$

Thus, this proves that

$$\|v_{CM}\|_{m_{TM},\Delta t} < \|v_1\|_{m_1,\Delta t} + \|v_2\|_{m_2,\Delta t} \tag{2.55}$$

is a norm, therefore it induces a distance in the vector space of the velocities[3].

2.5.3 Additivity of Actions

In order to allow for the additive nature of the GAI in (2.39) and (2.40), we need to show the additivity of Actions. We prove the additivity of Actions by first starting off

---

[3]Although it can be shown to apply to situations where the potential is not zero, the exact nature of this distance needs to be verified and explored further.

34

by computing the S Metric for two objects by constructing the combined Action for the two objects, S_{12}. Again under the assumption of U=0, we start off by using the S for one object, as shown in (2.6). From this, we compute the Action for both objects by first constructing their Lagrangian:

$$L_{12} = \frac{1}{2}m_1 v_1^2 + \frac{1}{2}m_2 v_2^2 \tag{2.56}$$

This leads to a combined Action for the two objects:

$$
\begin{aligned}
S_{12} &= \int_{t_a}^{t_b} L(q,\dot{q},t)dt \\
&= \int_{t_a}^{t_b} \frac{1}{2}m_1 \left(\frac{x_{1,b}-x_{1,a}}{t_b-t_a}\right)^2 + \frac{1}{2}m_2 \left(\frac{x_{2,b}-x_{2,a}}{t_b-t_a}\right)^2 dt \\
&= \frac{1}{2}m_1 \frac{(x_{1,b}-x_{1,a})^2}{t_b-t_a} + \frac{1}{2}m_2 \frac{(x_{2,b}-x_{2,a})^2}{t_b-t_a} = S_1 + S_2
\end{aligned} \tag{2.57}
$$

Thus showing the combined Action is just the sum of the individual Actions:

$$S_{12} = S_1 + S_2 \tag{2.58}$$

where S_{12} is used as the S Metric for composite systems.

Chapter 3

Neurobiologically-Inspired Statistical Integration Mechanisms

3.1 Introduction

Existing neurobiological models for motion recognition do posit the existence of a coupling or integration of the two pathways but they leave any specific mechanism to combine the two pathways as an open question [23, 22]. In this section, we propose a complex computational model for the integration of the two pathways based upon object-recognition models from neurobiology and neuromorphic computing which we represent using three different computational processes for this integration: Total Integration (TI), Partial Integration (PI), and Weighted Integration (WI). One of our proposed mechanisms, WI, does no worse than either of the two pathways individually. To the best of our knowledge, this is the first ever method that proposes a computational equivalent to the integration of the neurobiological models of motion recognition and which is applied to real world data.

Some researchers [21, 24] suggest the motion pathways integration is similar to object recognition; and since others [31, 56, 50] studying image-based recognition have been inspired by the success of biologically-motivated approaches for object recognition, we are similarly proposing the application of computational models that have proven effective for object recognition to motion recognition. While biologically-inspired approaches like those used by [31, 56, 50] have validated this approach, the success of non-biologically-motivated systems for extending object recognition descriptors to actions [16] also lend support to such a thrust.

In particular, neuromorphic computing [47, 45, 58] builds computational models for object recognition motivated by neurobiological pathways. Building upon this and recent work in the neurobiological community which shows the dorsal and ventral processes could be integrated through a process of *feature integration* [64] or *biased competition* [14, 6, 35, 69, 13] as originally outlined by [15, 52], we propose a computational model for the fusion of the motion energy and form/shape pathways, as shown in Figure 3.1. We employ a variant of these ideas using a statistical hypothesis testing framework with the bootstrap, creating Total Integration (TI), Partial Integration (PI), and Weighted Integration (WI) models. One advantage is that this allows us to also implement a variation of the Maximum-Like operation utilized in each of the pathways of motion recognition [23, 59] as well as for the fusion, itself. A Maximum-like operation is one that results in an output signal that approximates the maximum among several input signals [23, 22].

The integration mechanism has parallels to a lot of work in activity recognition that combines motion and form information; e.g., motion and form integration is done in Shape Dynamical Models and Bag of Video Words [66]. Thus, shape trajectories [68], bag

Figure 3.1: Computational integration of the two pathways in our framework

of video words [41], Mach filter [53], and actions as shapes [26] all make use of motion and form, either implicitly or explicitly. However, each of them requires a separate set of heuristics to combine motion and shape. In our approach, we use tracks directly to first analyze the motion information and then analyze the informaion in the sequence of images for the form components. *Also, first using motion energy and then form/shape allows for a hierarchical classification whereby the motion can perform a gross, top-level classification that can be computationally more efficient than the greater granularity of a detailed form exploration of the form; this is a great advantage when searching large video databases, for example.* Since our representation uses the abstract Hamiltonian framework and is based upon the physical motion information of the system, garnered directly from the tracks, it is much more physically significant as compared to "energy-based" methods which take some property of the image, like pixel intensity, and square it or cast it as a squared distribution or find its fourier energy [39, 47].

3.2 Overview of Proposed Framework for Motion Recognition

Our proposed framework for motion recognition is summarised in Figure 3.2. As can be seen there, the task we have is to take a probe/test query, containing the motion of a subset of the objects from the gallery/database, and match the motion of each object in the probe to the gallery/database. We start off by computing the HES curves for the Motion Energy Pathway for each object in the probe. Simultaneously, we compute the shape information for the Form Pathway for each object in the probe. These are then compared, individually, with each object in the gallery. These normalized similarity measures are then sent to the Integration module which can be done using different methods (e.g., Weighted Integration using the bootstrap, etc.).

3.3 Hypothesis Testing and the Bootstrap

Hypothesis testing can be considered a five-step process, given a set of distance scores of a probe sequence against all elements of the gallery. Below, we outline the process for our application.

1. Establish a null hypothesis, H_o, and an alternative hypothesis, H_a. In our case, the null hypothesis would be that a distance measure is not significant while the alternative would be that it is.

2. Establish a significance level, α, which is usually set to 0.05 [37].

3. Collect the data, select the test statistic, and determine its value (observed) from the
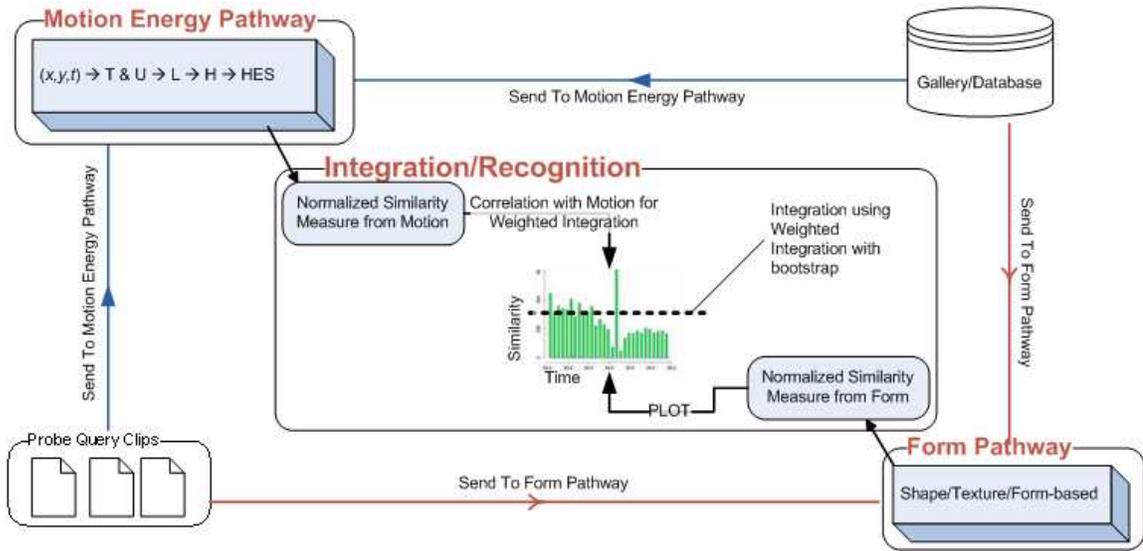
Figure 3.2: Proposed Framework for motion recognition by searching a database for a query: final recognition decision is made in the Integration module

sample data (in our case, this is creating the distance matrix).

4. Determine the criterion for acceptance/rejection of the null hypothesis by comparing the observed value to the critical value. In our case, this critical value threshold is determined via the appropriate Confidence Interval. The 2-sided Confidence Interval will have a lower critical value of the 0.025 quantile and an upper critical value of the 0.975 quantile for $\alpha = 0.05$. In our implementation, we use the bootstrap to find the variance of these quantiles (please see below for details of the bootstrap and confidence intervals).

5. Conclusion: Reject the null hypothesis if the observed value falls within the critical region (i.e., falls outside the Confidence Interval determined by the quantiles). In our case, the null hypothesis would be that all quantiles are equally significant and the alternative hypothesis would be that at least one quantile is different (i.e., is

statistically significant); these significant quantiles would be the ones that fall in the critical region.

3.3.1 Bootstrap

Following the work in [19, 72, 73], we use the bootstrap to find the variance of the desired quantile threshold used within the various Integration models. Bootstrap is a nonparametric method which lets us compute some statistics when distributional assumptions and asymptotic results are not available. In statistics, it is more appealing to compute the two sided α significance threshold (confidence interval) via bootstrapping because of its accuracy and lack of assumptions.

A confidence interval is a range of values that tries to quantify the uncertainty in the sample. A confidence interval can be two-sided or one-sided, as shown in Figure 3.3; e.g., the 95% 2-sided confidence interval shows the bounds within which one can find 95% of the population (similarly for the 1-sided upper and lower confidence bounds). Confidence intervals are also equivalent to encapsulating the results of many hypothesis tests; if the confidence interval doesn't include H_o, then a hypothesis test will reject H_o, and vice versa [37]; in fact, both confidence intervals and hypothesis testing are key elements of inferential statistics. This is important in our method as we utilize a hypothesis testing framework, within which we use the bootstrap to estimate the confidence intervals.

The bootstrap works by re-sampling with replacement to find the variance of a statistic on a sample, as shown for our specific case in Table 3.1. We may use this algorithm twice, depending on the Integration variant we're computing: once for the upper quantile and once for the lower quantile. One way to estimate confidence intervals from bootstrap
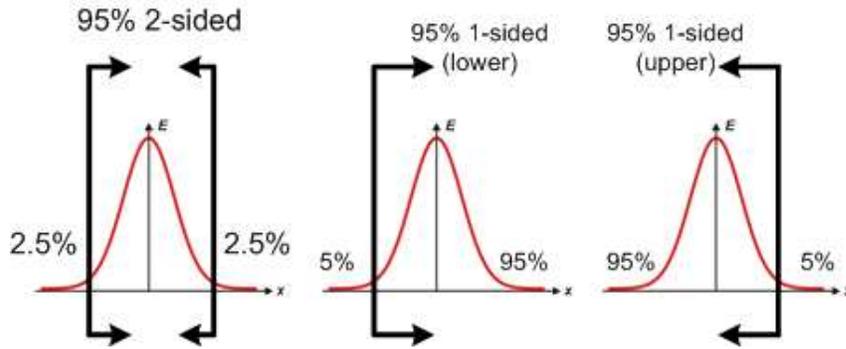
Figure 3.3: 2-sided and 1-sided Confidence Intervals (CI): the first diagram shows a 2-sided CI showing the confidence interval in the middle and the critical regions to the left and right; the second diagram shows a 1-sided lower bound with the critical region to the left; the final diagram shows a 1-sided upper bound with the critical region to the right; the E just indicates the mean expectation value.

samples is to take the α and $1 - \alpha$ quantiles of the estimated values, called bootstrap percentile intervals. For example, for the upper quantile, this confidence interval would then be given as $CI = (q^u_{lower}, q^u_{upper})$, with $lower = \lfloor N\alpha/2 \rfloor$ and $upper = N - lower + 1$, where N is the number of bootstrap samples and $(q^u_{lower}, q^u_{upper})$ are the lower and upper critical values of the bootstrap confidence interval bounds.

So, in our case, we use the hypothesis testing framework to establish the critical region quantiles for the Confidence Interval associated with our significance level, α, for each probe in the distance matrix. In order to find the variance of the desired quantiles (both lower and upper), we use the bootstrap method from Table 3.1. We use the same significance level, α, as before and derive the bootstrap critical region, $CI = (q^u_{lower}, q^u_{upper})$, for the upper quantile and $CI = (q^l_{lower}, q^l_{upper})$ for the lower quantile. We also use the alternate method (using just the mean of the quantile threshold) from the bootstrap for comparison.

Step 1: *Experiment*: Get DistanceMatrix

$DM = 2.41, 4.86, 6.06, 9.11, 10.20, 12.81, 13.17, 14.10, 15.77, 15.79$ (these numbers are randomly generated numbers for illustrative purposes only as the actual distance matrices are too big to list in this example).

Step 2: *Re-sample*:

Using a pseudo-random number generator, draw a random sample of $length(DM)$ values, with replacement, from DM. Thus, one might obtain the bootstrap resample

$DM' = 9.11, 9.11, 6.06, 13.17, 10.20, 2.41, 4.86, 12.81, 2.41, 4.86$.

Note that some of the original sample values appear more than once, and others not at all

Step 3: *Calculate the bootstrap estimate*: Compute the *bootstrap percentile intervals* $(1 - \alpha)100\%$ for either the upper or the lower quantile for the resample; in this example, we might be computing the upper quantile and get $q_1^u = 13.008$ for the upper quantile of the first bootstrap run (if we were computing the lower quantile, we would get q_1^l for the lower quantile computed for the first bootstrap run).

Step 4: *Repetition*: Repeat Steps 2 & 3 N (say 1000) times to obtain a total of N bootstrap estimates: $q_1^u, q_2^u, \ldots, q_N^u$ and sort in increasing order (if we were computing the lower quantile, we would instead have $q_1^l, q_2^l, \ldots, q_N^l$).

Step 5: *Parameter estimation*: The desired quantile threshold is then derived from the bootstrap confidence interval, which is given as a 2-sided confidence interval:

$CI = (q_{lower}^u, q_{upper}^u)$, where $lower = \lfloor N\alpha/2 \rfloor$ and $upper = N - lower + 1$ (if we were computing the lower quantile, the bootstrap confidence interval would instead be $CI = (q_{lower}^l, q_{upper}^l)$, where $lower = \lfloor N\alpha/2 \rfloor$ and $upper = N - lower + 1$). Alternatively, some methodologies use the mean of $q_1^u, q_2^u, \ldots, q_n^u$ for the desired quantile threshold of the upper quantile and the mean of $q_1^l, q_2^l, \ldots, q_n^l$ for the desired quantile threshold of the lower quantile.

Table 3.1: Outline of Bootstrap Quantile Analysis
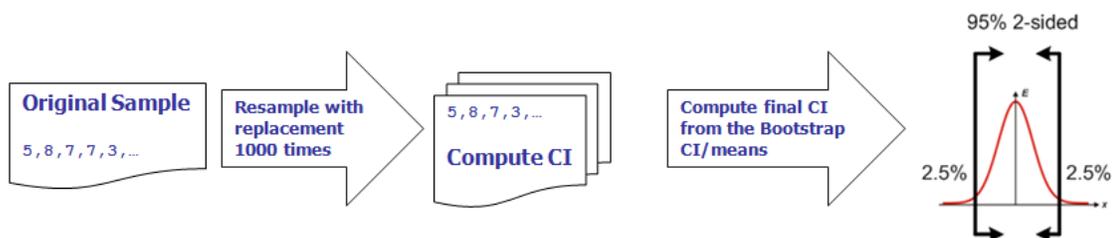
Figure 3.4: Overview of Bootstrap. This figure shows how the original sample is re-sampled (with replacement), say, 1000 times. In each re-sampling, a Confidence Interval is computed based on that sample. Eventually, the final Confidence Interval is estimated from either the Bootstrap Confidence Interval (on the CI computed on each re-sample) or the means (again, of the CI computed on each re-sample).

3.4 Integration strategies

3.4.1 The Form Pathway: HOGs and Shape-based Features

Since the Form Pathway is posited to have orientation detectors and also recognizes body shapes and color [22], we use well-established methods in machine vision to calculate exactly these features in order to develop its computational representation. Our construction provides flexibility on the Form Pathway since new approaches in low-level feature extraction can be employed easily within our framework and the appropriate Form Pathway features can be used. For the present work, we use Histogram of Oriented Gradients (low-resolution experiments) and shape features with DTW (high-resolution and gait experiments). Also, since we are analyzing video, we will need to consider the shape/gradients over a sequence of frames.

For modelling the sequence of shapes for an activity, we used the methodology from [68]. It presents an approach for comparing two sequences of deforming shapes using both parametric models and nonparametric methods, where we use the latter. In this implementation, Kendall's definition of shape is used for feature extraction. Since the shape feature rests on a non-Euclidean manifold, they propose a modification of the Dynamic time-warping algorithm to include the nature of the non-Euclidean space in which the shape deformations take place. They apply this algorithm for gait-based human recognition on the USF dataset by exploiting the shape deformations of a person's silhouette as a discriminating feature and they also provide results for motion recognition.

For the low-resolution case, we relied upon a variation of the standard Histogram of Oriented Gradients [12] which counts occurrences of gradient orientation in localized

portions of an image on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved performance.

3.4.2 Integration Mechanisms

The usual Neuromorphic Computing tack is to integrate the dorsal and ventral pathways via the Integration module, usually by weighting them, as shown in Figure 1.1, above. We propose a computational approach to implement the different Integration variants within a Hypothesis Testing framework in which we also use the bootstrap [19, 72, 73] to ensure reasonable limits.

3.4.3 Integration Variants

Given a set of distance scores of a probe sequence against all elements of the gallery, Hypothesis Testing lets us choose between the motion and form features or come up with a combination of them, with the bootstrap being used to find the variance of the quantiles on the sample. After a sample is collected from an experiment, we can calculate a statistic on it (like the mean or quantiles, for example), and then use the bootstrap to calculate the variance in that statistic, (e.g., via a Confidence Interval, CI, as shown in Figure 3.3).

Neuromorphic computing approaches have examined different integration methodologies, including simple pointwise multiplication, as well as exploring more standard neurobiological integration mechanisms such as *feature integration* [64], in which simple visual features are analyzed pre-attentively and in parallel, and *biased competition* [15, 52], which "proposes that visual stimuli compete to be represented by cortical activity. Competition may occur at each stage along a cortical visual information processing pathway. The out-

come of this competition is influenced not only by bottom-up, sensory-driven, activity but also by top-down, attention-dependent, biases".

Our Integration approach is a variant of these different methods and, in this work, we develop three different computational models for this integration: Total Integration (using a 1-sided upper bound CI), Partial Integration (2-sided CI), and Weighted Integration (2-sided CI).

Total Integration (TI): In this case, only the winners survive. First, the observed distance matrix is converted into a similarity matrix; then, the bootstrap quantile analysis (shown in the Appendix) is done on both the Motion measures and the Form measures for the upper quantile only. If the value of either the Form or Motion is lower than its upper bootstrap quantile analysis confidence bound, then its value is set to 0; if both are higher than their upper bootstrap quantile analysis, the resultant value is set to the pointwise correlation between the normalized Motion and Form measures (only the values which "survive" in both are returned as the final result).

Partial Integration (PI): Here, a 2-sided CI is used in which the observed distance measure is lowered to 0 if it is less than the lower distance quantile or changed to the max value if it is greater than the upper distance quantile; intermediate values are set to the pointwise correlation between the normalized Motion and Form measures.

Weighted Integration (WI): The Form values are weighted based on the Motion values; if the observed distance value of the Form and the Motion is lower than the lower distance quantile obtained from the bootstrap quantile analysis for both, then the value is set to 0; if either is higher than the upper quantile analysis, it is set to the max value; all other values are set to the unaltered Form value. In this way, WI ensures that it always

does no worse than the Form.

For future research, we are considering other integration mechanisms, including using MCMC approaches adapted for our Hamiltonian dynamics model. In fact, our framework allows us to use any model for the Integration component that might be appropriate, just as in the Form Pathway component. However, the details of a particular implementation might affect the accuracy of the results but not the fundamental Integration principle since the method is not linked to the framework. However, these variations are a separate, future piece of work.

3.5   Results of the Integration

Since the original neurobiological model for motion recognition does not, as of yet, posit an integration mechanism, one of our main contributions is to follow through with the conjecture that the same sort of integration happens in the motion recognition pathways as in the object recognition pathways. We thereby propose a complex integration model based upon neurobiological and neuromorphic computing models for object recognition.

In the experiments that follow, we show that the Integration mechanism helps reduce the search space (using the Weizmann dataset) and also helps with overall recognition when either the motion or the form (or both) pathways fail or underperform. We show how, in the USF Gait dataset, although the form model performs well, when we integrate that with the motion energy computational model, it improves the overall performance; although both do reasonably well on their own, the integrated version does better than either alone. Finally, we also demonstrate the utility of the HES as a discriminating function of the computational

motion energy pathway and the ability of the Integration to reduce the search space.

3.5.1 Experimental Background

For all of these experiments, tracking and basic object-detection was already available [33] and we utilized these (x,y,t) tracks to compute the Lagrangian and Hamiltonian, following our development in Section 2.2.2. We used high-resolution and low-resolution video from the Weizmann dataset. For the low-resolution case, we further used the tracks to get the Histogram of Oriented Gradients for our Form Pathway whereas, in the high-resolution case, we utilized shape (as defined in [68]) for the Form component. The histogram of similarities in each time window was computed using standard methodologies, as described below. We then utilized Hypothesis Testing with bootstrap to derive the threshold for peaks in the distributions that might compete for selection/matching.

3.5.2 Integration helps reduce search space

We now show how integration helps reduce the search space. In this case, we demonstrate on the Weizmann dataset. The Weizmann dataset (`http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html`) consists of a database of 90 low-resolution (180 x 144, deinterlaced 50 fps) video sequences showing nine different people, each performing 10 natural actions. We analyze these using both shape methods [68] (as discussed in Section 3.4.1), as well as via the GAI. Using both procedures, we see the resulting similarity matrices in Figure 3.5 (a) and (b), respectively. Finally, in Figure 3.5 (c), we see the result of integrating via WI. In each of the distance matrices, both axes consist of the people grouped by the activity: bend, jack, jump, pjump, run, side, skip, walk, wave1, wave2. So the first

a)

b)

c)
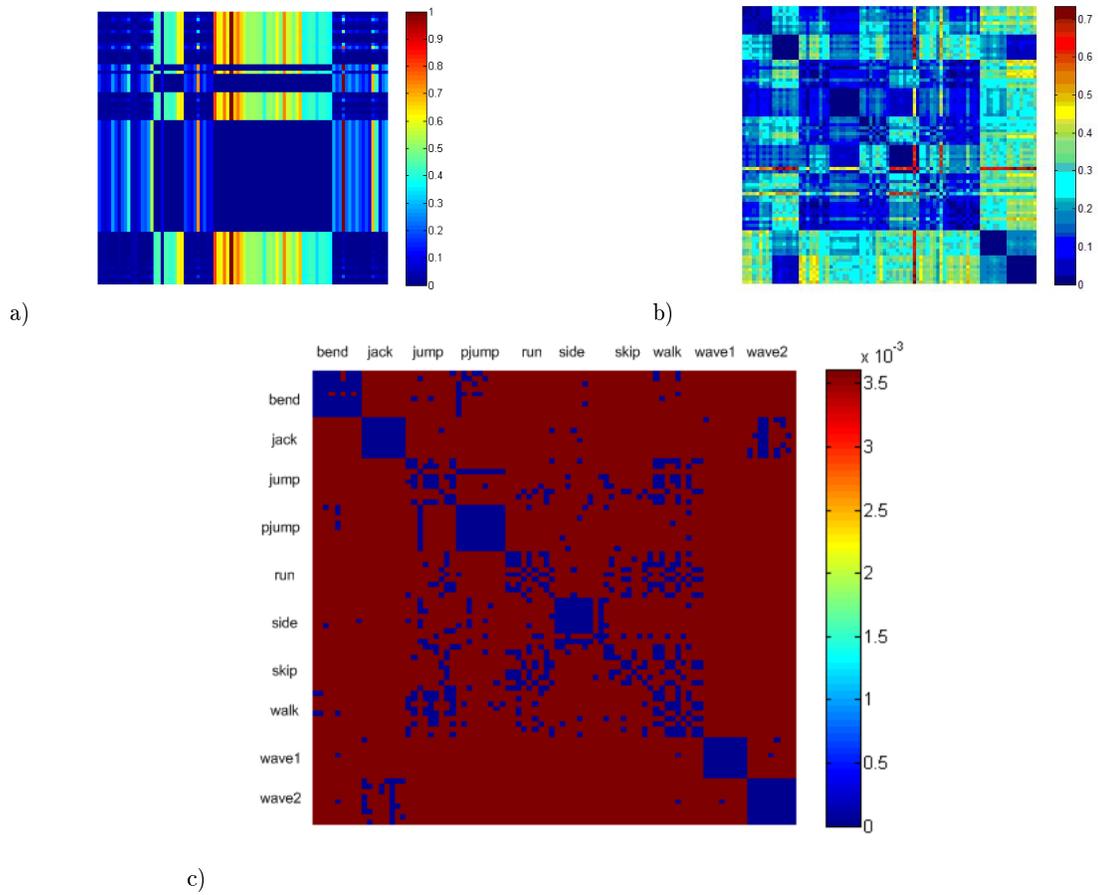
Figure 3.5: Similarity matrices on the Weizmann dataset for a) GAI only, b) Shape Methods only, and c) Integration using WI. Both axes consist of the people grouped by the activity: bend, jack, jump, pjump, run, side, skip, walk, wave1, wave2. So the first nine rows are each person bending, the next nine rows are each person doing a jumping jack, etc. In (c), we see the result of integrating via WI. As can be seen in the matrices, WI combines both pathways in such a way as to do no worse than either pathway by itself.

nine rows are each person bending, the next nine rows are each person doing a jumping jack, etc. This clustering by the different methods is shown explicitly in Figure 3.6, where we see the Motion pathway correctly isolates pjump and jump; the Form pathway further clarifies bend, jack, side, and run; finally, Integration discerns wave1 and wave2, with skip and walk remaining grouped.

As can be seen in the matrices in Figure 3.5 and the diagrams in Figure 3.6, GAI alone, in Figure 3.5 (a), groups together bending and jumping jacks; partially groups the jumping sideways; fully groups jumping in place; confuses running, galloping sideways, skipping, and walking; and confuses waving 1 and waving 2. Form alone, in Figure 3.5 (b), groups bending and jumping jacks correctly; partially groups the jumping sideways; fully groups jumping in place; partially groups running; partially groups galloping sideways; confuses skipping and walking; and partially confuses waving 1 and waving 2.

The Integration, however, in Figure 3.5 (c), does better than both in most cases and no worse than the better method, form, in all cases. As can be seen, it groups bending and jumping jacks correctly; partially groups the jumping sideways; fully groups jumping in place; partially groups running; fully groups galloping sideways; confuses skipping and walking; and fully groups waving 1 and waving 2.

In order to demonstrate the benefit of the Integration mechanism provided by our framework, we tested cases where the HES failed, as well as cases where the Form approach failed, with experiments on the Weizmann dataset to compare both to Integration in Table ??. We thus show that the integrated combination works better than using only one source of information.

50
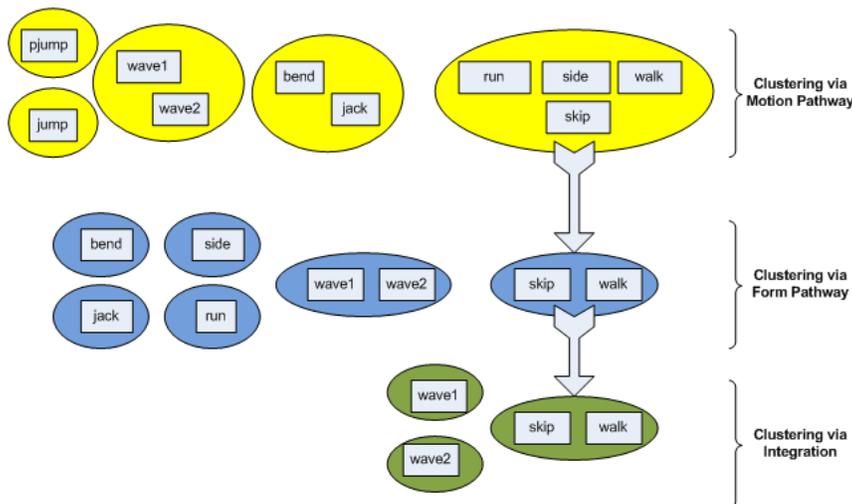
Figure 3.6: Data Clustering via Motion Pathway, then Form Pathway, and finally Integration. As seen here, the Motion correctly isolates pjump and jump; Form further clarifies bend, jack, side, and run; finally, Integration discerns wave1 and wave2, with skip and walk remaining grouped.

3.5.3   Integration improves performance of gait recognition

We show how, in the USF Gait dataset, although the form model performs well, when we integrate that with the motion energy computational model, it improves the overall performance. We experimented with videos from the standard USF gait dataset consisting of 67 people walking on different surfaces (grass and concrete) with different shoe types and different camera views. The HES curve generated for each person is actually a multi-dimensional vector composed of HES curves for all the points on the contour of that person's silhouette, as shown in Figure 2.3. The form component was calculated using the shape of the silhouettes and computing similarity using DTW in the shape space.

We utilized WI to bias the Form component with the Motion component and then used the bootstrap to set the threshold for peaks in the distributions that might compete for selection/matching. The results are plotted as both distance matrices as well as Cumulative

Match Score (CMS) graphs, which plot probability vs. rank; results are in Figures 3.7 and Figure 3.8. We also see the Integration approach consistently outperforms the Form Pathway approach alone, as seen in Figure 3.8. The singular exception is Probe B in rank 1; this is because WI favours the Form method more heavily than the Motion Energy Pathway method and, in this case, the Form method misses the real match and guesses matches that are far removed from the real match, as seen in the similarity matrix in Figure 3.7. Please note that although these results are specific to our Form approach, it is expected that similar improvements would be realized using other approaches.
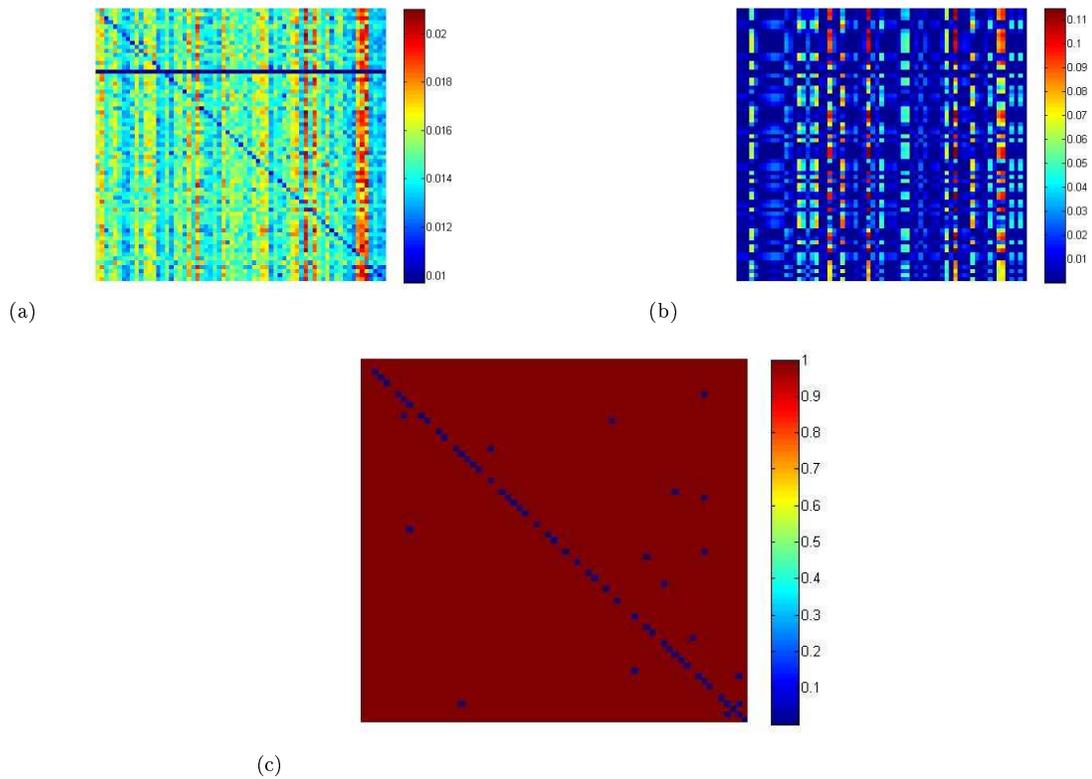
Figure 3.7: Similarity Matrices for USF Gait dataset examined using (a) Form Pathway, (b) Motion Pathway, and (c) the WI Integrated Framework on Probe A for all seven probes in the USF Gait. Although the form model performs well, when we integrate that with the motion energy computational model, it improves the overall performance as seen by the matching in (c). The overall CMS matching is shown in (d).
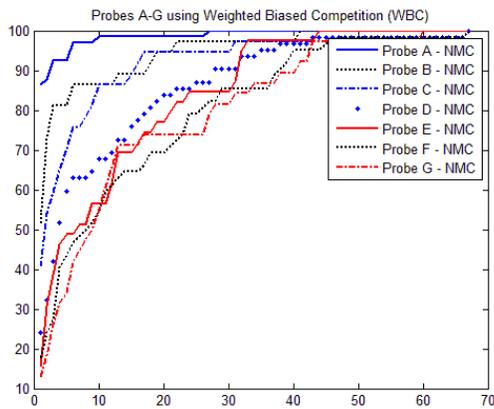
(a)

Probe	Rank 1		Rank 5	
	Form	Integ	Form	Integ
A	81.8	86.4	92.4	92.4
B	59.5	51.4	81.1	83.8
C	40.5	40.5	70.3	70.3
D	21.0	24.2	54.8	58.1
E	15.4	15.4	46.2	46.2
F	16.1	17.7	41.9	43.6
G	13.2	13.2	34.2	34.2

(b)

Figure 3.8: CMS Curves in (a) and Comparison of Form and Integration Rank 1 & Rank 5 match probabilities on USF Gait (b).

Chapter 4

Data Driven Hamiltonian Monte Carlo

4.1   Introduction

The dynamic nature of video makes activity recognition from video databases a very difficult problem. It requires an analysis of both the motion and the image features of the system being studied in video. These often disparate features need to be combined in order to realize optimal recognition or classification. Their integration, however, leads to an even greater problem as the final search space, which is composed of the combined motion and image spaces, is usually enormous. In this work, we present a new approach, the Data Driven Hamiltonian Monte Carlo (DDHMC), which extends the Hamiltonian Monte Carlo (HMC) and allows us to simultaneously search over the combined space in a concerted and efficient manner using well-known Markov Chain Monte Carlo (MCMC) techniques.

Monte Carlo	— A general stochastic technique that uses random numbers to examine some problem. — More specifically, Monte Carlo is a sampling method for iteratively evaluating a deterministic model using sets of random numbers as inputs where the samples are drawn from a probability distribution. — The general idea is to use Independent and Identically Distributed (iid) simulations of a random process to approximate expectations by sample averages.
Markov chain	— A Markov chain can be thought of as a finite state machine with probabilities for each transition. — More specifically, it is a stochastic process that consists of a finite number of states with probabilities for transitions from each state to the next and having the property that future states depend only on the present state. — E.g., p_{ij} is the probability of moving from state i to state j.
MCMC	— Markov Chain Monte Carlo (MCMC) is a way to generate a random sequence of values for parameter x from a target probability density function, $\pi(x)$; all of the standard MCMC techniques assume that you start off with some target distribution that you can evaluate but cannot sample from. — MCMC uses simulations of a Markov chain to approximate expectations by sample averages. — It essentially replaces IID simulations in Monte Carlo with Markov chain simulations and simulates a Markov chain whose invariant states follow a target probability in a very high dimensional state space.
HMC	— Hamiltonian Monte Carlo (HMC) is an alternative MCMC technique in which an auxiliary (fictitious) momentum variable is introduced for each parameter of the target probability distribution function (pdf). — It thus simulates a physical system by creating a Hamiltonian, H, where the potential function is the target density and uses Hamiltonian Dynamics to explore a new pdf (which is proportional to e^{-H}) by drawing random samples from it. — HMC tries to avoid the random walk behavior of regular MCMC and allows proposals to move across the sample space in larger steps, thus allowing the proposals to be less correlated and converge to the target distribution more rapidly.
Hamiltonian Dynamics	— Hamiltonian Dynamics (or Mechanics) is an elegant and powerful alternative formulation of classical mechanics that not only gives the equations of motion for a system but, more importantly, provides greater, and often more abstract, insight about the system. — It provides a framework based upon the Principle of Least Action that can be extended to all other laws of physics (in fact, all fundamental laws of physics can be expressed in terms of a least action principle). — Hamilton's equations, using the Hamiltonian H, are equivalent to the Euler-Lagrange equation; Hamilton's equations are primarily of interest in establishing basic theoretical results, rather than determining the motions of particular systems.
DDHMC	— A physically driven method to address the problem of activity recognition by combining the usual energy-based Hamiltonian approach of a Traditional HMC with data-driven proposals derived from video observations. The resulting approach is what we call the Data Driven HMC (DDHMC). DDHMC is a potentially general approach but we mainly demonstrate it in the context of activity recognition in the current work

Table 4.1: A summary of some of the basic terms used in this work

56

4.1.1 Hamiltonian Monte Carlo (HMC)

Traditional MCMC techniques have been widely used in many disciplines to simulate a Markov Chain whose invariant states follow a target probability distribution in a very high-dimensional state space; they do this by generating fair samples from an easy-to-sample proposal probability distribution [42] (fair samples show us what states are typical for the underlying system). MCMC is normally used for simulation, integration (computing integrals in very high-dimensonal space), or global optimization and Bayesian inference [46]. Normal MCMC techniques are usually based on stochastic sampling algorithms (e.g., Gibbs sampling); as a result, they can be very slow and inefficient when sampling high-dimensional parameter spaces. In order to make these algorithms more efficient, [17, 18] originally introduced the Hamiltonian Monte Carlo (HMC) in the context of solving problems in Quantum Chromodynamics.

HMC is an MCMC technique that uses gradient information to make traditional MCMC more efficient by leveraging the advantages of Hamiltonian dynamics to investigate how the system evolves in parameter space (as explained later, this is the phase space). This gives the HMC higher acceptance rates, less correlated and faster converging chains, and suppresion of the random walks in traditional MCMCs [4, 46]. Initially developed for investigating computer simulation of lattice field theory/Quantum Chromodynamics using an artificial dynamics [18, 46], HMC has also been applied to simulation of classical physical systems, probabilistic reasoning and statistical inference, and parameter estimation [4].

In general, the HMC is faster than classical stochastic sampling-based (Gibbs sampling, Metropolis-Hastings, etc.) optimization algorithms. By following the dynamical path

in phase-space, we can propose candidate moves that are far away from the current state but that still have a substantial chance of being accepted. This gives us a way to efficiently explore large regions of phase-space by simulating Hamiltonian dynamics in fictitious time in the traditional HMC. The benefit of following the trajectory of the system in phase-space is that it eliminates the random walk aspect of the chain while also improving mixing and producing more accurate estimates and allowing us to explore quickly regions that are far away from the current state. Thus, (the analysis of motion via) Hamiltonian dynamics is the key to the approach proposed in this work.

4.1.2 HMC Extension and Application to Activity Recognition

Besides the HMC, another recent innovation in the development of MCMC was the DDMCMC, developed by [71], which uses data-driven proposals to make the Markov Chain efficient. Although MCMC techniques have been applied in various activity recognition applications, neither HMC nor DDMCMC have been used for activity recognition. DDMCMC has mainly been applied to image segmentation and object recognition [65, 71]; similarly, although HMC has been applied to particle filters and tracking [10, 49, 61], these techniques have never been applied to activity recognition to the best of our knowledge.

In this work, we form the logical next step in HMC development by introducing the DDHMC, which uses data-driven proposals to make the search more efficient. In addition, we apply the HMC and DDHMC framework to activity recognition for the first time. Almost all activity recognition methods use some variant of motion plus form analysis but utilize different heuristics to conduct that analysis; some typical examples being shape trajectories [68], bag of video words [41], Mach filter [53], and actions as shapes [26], all of which make use

58

of motion and form, either implicitly or explicitly. Thus, the integration afforded by DDHMC provides a stochastic framework that is especially suited for activity recognition. We propose two DDHMC frameworks: one in which we first analyze motion in phase space using the basic tracks and then integrate in the form (image characteristics like shape, gradients, etc.) methodology via the HMC framework; and another in which we first analyze form-based information and then integrate in the motion energy information via the HMC framework. Our proposed approach, using motion plus form information, thereby provides a natural framework for the integration of the two and brings the robustness of statistical methods to activity recognition.

4.1.3   The Motion Space

Motion, in fact, underlies all activities; human activities are defined by motion. For the last 450 years, the exacting study of motion has been the cornerstone of physics, over which physicists have unlocked its deep, underlying structure. This study of motion theory can be helpful in the modeling and recognition of human motion. In order to study activity recognition, we start with a rigorous study of motion using ideas grounded firmly in fundamental physics. From first principles, we develop a method to extract information about the motion of the underlying physical systems we consider in video.

Using this physics-based methodology, we derive, in the attached appendices and in Section 4.3.2, the Hamiltonian equations of motion for various objects (either entire objects or the parts of a single object) involved in activities observed in video.

We can easily visualize this Hamiltonian for simple cases, like the exchange activity in Figure 2.2(a), where we see two people exchanging a box. Plots of the Hamiltonian

equations of motion, as explained in Section 4.3.2, can give us a sense of the energies associated with this activity, both in the idealized case (Figure 2.2(b)) and for the experimentally observed case (Figure 2.2(c)).

As another example, we can see two cars following each other in Figure 2.3. Here, the first car, whose trajectory is labeled in orange, is the lead car and executes a U-turn; the second car, trajectory in blue, follows it and also makes a U-turn, whereas the third car, whose trajectory is in red, follows it for a while and then turns away. We see the similarities and variations in the Hamiltonian (phase-space) plots.

4.1.4   The Integration Approaches

We used this physically-significant Hamiltonian, along with well-known form- or shape-based methodologies for the image space, to yield a Data Driven HMC (DDHMC), which can efficiently explore the combined motion energy space and image space. An overview of the main differences between a traditional HMC and our novel DDHMC algorithms is highlighted in Table 4.2 and the rest of this work explores these differences in detail.

We propose two different approaches to this integration of form and motion energy information: one whose data-driven component is based on image proposals and another whose data-driven component is based on motion energy proposals. We develop both variants because, depending on the specific application or problem, either the form or the motion might outperform the other. Since our integration affords a hierarchical classification scheme in which the data-driven proposal does an initial, gross classification, we create both variants to take best advantage of whichever method is most suited to the finer granularity of

60

classification; e.g., in image analysis, the motion-based proposals give a gross classification and the image-based confirmation via the traditional HMC architecture yields the final, finer classification in the integration afforded by the DDHMC framework.

In brief, we achieve the integration by changing the 1st step in a Traditional HMC: instead of sampling from a Normal distribution, we use a data-driven proposal. The Hamiltonian dynamics in the 2nd step and the Metropolis-Hastings in the 3rd step are also slightly modified, as explained in detail in Sections 4.4.3 and 4.4.1.

The integration of a Hamiltonian energy-based approach with an image-based data-driven proposal allows us to simultaneously search over both the motion energy space and image space in a concerted manner, unlike traditional HMC methodologies which only address a single space, thus reducing the enormity of the search space. In addition to our data-driven innovation, we are proposing, to the best of our knowledge, the first such application of our novel HMC algorithms to the problem domain of activity recognition in video, taking into account the image analysis results and using the physical motion information of the system.

4.1.5 Contributions

In this work, we propose a physically driven method to address the problem of activity recognition by combining the usual energy-based Hamiltonian approach of a Traditional HMC with data-driven proposals derived from video observations. The resulting approach is what we call the Data Driven HMC (DDHMC). DDHMC is a potentially general approach but we mainly demonstrate it in the context of activity recognition in the current work, where we propose to use the DDHMC, on top of human feature tracks (e.g.,

Step	Traditional HMC	Data-Driven HMC Approach - Image Proposals	Data-Driven HMC Approach - Motion Proposals
1.	Sample from a Normal Distribution	Sample from Data-Driven Gibbs Distribution, based in image proposals	Sample from Data-Driven Gibbs Distribution, based on motion proposals
2.	Dynamic Transition (Leapfrog)	Dynamic Evolution (compute the difference of Hamiltonians along the trajectory)	Dynamic Transition (Leapfrog)
3.	Markov Chain using Metropolis-Hastings	Markov Chain using modified Metropolis-Hastings	Markov Chain using modified Metropolis-Hastings

Table 4.2: Comparison of the main loops of the Traditional HMC and Data-Driven HMC Approach

points on the human silhouette contour or on body parts in high-resolution or centroids in low-resolution), in order to classify human activities. These tracks may come from a user query or from averaging the Hamiltonian over all the training tracks (which do not include the test track).

We propose two different forms of the DDHMC: $DDHMC_{Motion}$, which uses motion-based data-driven proposals, and $DDHMC_{Shape}$, which uses shape-based data-driven proposals. Both variants of the DDHMC rely upon a data-driven component to make more informed proposals than the blind proposals generated within a Traditional HMC. These informed proposals, based on the likelihood of a particular track under a Kernel Density or Gibbs estimator, are then used as the data-driven portion of the HMC.

Thus, the initial classification of the activities based on the data-driven portion becomes the first step in a two-step, hierarchical classification scheme implemented by the DDHMC: the data-driven portion does a gross classification and then the Traditional HMC framework does a higher resolution classification with greater granularity. In $DDHMC_{Motion}$, we use a physically-significant Hamiltonian, derived from the tracks, to get the similarity distribution that will help guide the Traditional HMC framework. For $DDHMC_{Shape}$, we use a shape-based similarity distribution to help guide the physically-significant Hamiltonians derived from the tracks that are used within the Traditional HMC framework.

In order to derive these physically-significant Hamiltonians from the tracks, we start from first principles; on an intuitive level, we develop a method to extract an abstract representation of the motion of the underlying physical systems we consider in video. Our method assumes that the video is segmented into objects and their motion is given; then, from the physical motion and location information of objects over time, we can use the abstract Hamiltonian framework to estimate the potential and kinetic energy of the system and derive an approximation to its Hamiltonian. In the applications proposed in this work, our standing hypothesis is that the energy pattern across time, represented via the abstract Hamiltonian framework, tends to be similar in different video clips of the same motion for given classes of activities. However, this is not usually sufficient for activity classification, hence we need to analyze the image space, as well.

Therefore, our main contributions are:

- Creation of a data-driven framework for the HMC, thereby yielding a DDHMC, which allows for hierarchical classification.

- Application of the HMC architecture, via the DDHMC, to activity recognition for the first time.

- The development of two different data-driven proposal algorithms using both the motion energy and the shape.

4.2 Hamiltonian Monte Carlo

Although there is some variety in the different kinds of HMCs, in general, all HMCs are, to some extent, a combination of Monte Carlo methods with a Dynamical simulation method (referred to as Molecular Dynamics in [42]), a deterministic procedure to integrate the Hamiltonian equations of motion. This combination of Monte Carlo and Dynamics allows HMC to overcome the shortcomings of both its constituents: the low acceptance rates of Monte Carlo (for small trial moves) and the insufficient temporal development of Dynamics simulations at continuum timescales [63]. Thus, the Dynamics framework specifies the system's possible moves while the Monte Carlo decides on the final moves that will generate the equilibrium population (the final Markov chain). Because the Monte Carlo samples from the phase space globally, it compensates for the relatively short Dynamics simulation lengths. HMC, in fact, builds upon the Molecular Dynamics/Langevin algorithm as a means of simulating physical systems [17].

HMC does so using dynamic sampling and stochastic dynamics. Dynamic Sampling is based on a physical analogy: the gradient of the potential energy for a physical system gives the "force" that then acts to change system's configuration via momentum. This dynamical method is faster since it avoids the random walk element of traditional MCMC approaches.

Thus, even if the system under study is not a physical system, it is advantageous to invent an artificial dynamics for these non-physical systems, just to make it less random, as is done in traditional HMC. In the DDHMC, we avoid this problem further by using a data-driven paradigm, similar in spirit to the DDMCMC developed by [65], in order to help reduce the dimensionality of the search space.

In order to create an artificial dynamics, the usual approach in HMC is to introduce a fictitious "time", t; for a real physical system, t represents the actual physical time [46], as is true in the case of video analysis, where we use our DDHMC. In order to fully create this artificial (Hamiltonian) dynamics, the HMC also requires the creation of a fictitious momentum (for the Kinetic Energy) and an artificial Potential Energy. An independent extra set of momentum variables p with i.i.d. standard Gaussian distributions is usually introduced as the KE term in this fictitious phase space; i.e., p_i are sampled from $\mathcal{N}(0,1)$ just to give it a dynamical formulation whereas, for our case (and for the Molecular Dynamics described in [42]), the p_i have real significance since it is related to the actual momentum of the system under study [46] and can be computed from tracking in video.

For the potential energy, we start with the target probability distribution we want to sample from, $\pi(q)$. As long as this is continuous (so that we can calculate the derivatives), the Potential Energy is then represented as:

$$U(q) = -\lg \pi(q) \tag{4.1}$$

This log-likelihood then plays the role of a Potential Energy and this yields the full Hamiltonian (Energy) Function on a fictitious phase-space:

Lines	Analysis of Algorithm 1
1	Initialize the Markov Chain of phase space points we'll eventually return in Line 20
2-3	Start the main for loop
4-6	Draw an artificial momentum, p, from a normal distribution and initialize our current q^o to the previously accepted q, q_{i-1}
7-13	Let the system evolve dynamically (using the Leapfrog algorithm) from the initial point in phase space, (q^o, p^o), to the final point in phase space, (q', p').
14-16	Draw a random number, α, and compute the difference in Hamiltonians for the initial point and for the final point
17-22	If the difference is 0 or less, always accept the new phase space point, (q', p'), since this means the two trajectories intersected in phase space and so they have the same H; otherwise, accept it with probability α
23	Return the final Markov Chain of accepted phase space points, $\{p_i, q_i\}$

Table 4.4: Analysis of Algorithm 1

Algorithm 1 Traditional HMC Algorithm

Outline of the Traditional Hamiltonian Monte Carlo which builds a Markov Chain over q_i:

1: Initialize chain with $q_o, p_o, \Delta t$
2: // Start the outermost loop
3: **for** $i = 1$ to $nsamples$ **do**
4:    // Step 1. Sample from a Normal Distribution
5:    draw $p \sim \mathcal{N}(0, 1)$
6:    $(q^o, p^o) = (q_{i-1}, p)$
7:    // Step 2. Dynamic Transition Step using LeapFrog
8:    **for** $j=1$ to l **do**
9:       $p^{j-\frac{1}{2}} = p^{j-1} - \frac{\Delta t}{2} \bullet \nabla U(q^{j-1})$
10:       $q^j = q^{j-1} - \Delta t \bullet p^{j-\frac{1}{2}}$
11:       $p^j = p^{j-\frac{1}{2}} - \frac{\Delta t}{2} \bullet \nabla U(q^j)$
12:    **end for**
13:    $(q^{'}, p^{'}) = (q^l, p^l)$
14:    // Step 3. Final Metropolis-Hastings Step
15:    draw $\alpha \sim \mathcal{U}[0, 1]$
16:    $\delta H = H(q^{'}, p^{'}) - H(q^o, p^o)$
17:    **if** $\alpha < min\left(1, e^{-\delta H}\right)$ **then**
18:       $(q_i, p_i) = (q^{'}, p^{'})$
19:    **else**
20:       $(q_i, p_i) = (q_{i-1}, p_{i-1})$
21:    **end if**
22: **end for**
23: return $\{q_i, p_i\}_{i=0}^{nsamples}$

$$H(q, p) = -\lg \pi(q) + \frac{1}{2} \sum_i p_i^2 \qquad (4.2)$$

The canonical (joint) distribution associated with this Hamiltonian is $\pi(q, p) = \frac{1}{Z} e^{-H(q,p)}$ and, in this fictitious dynamics, the field of forces is thus supplied by the score function (i.e., the derivatives of the log-likelihood) [4]:

$$\frac{dq_i}{dt} = \dot{q}_i = \frac{\partial H}{\partial p_i} = p_i$$
$$\frac{dp_i}{dt} = \dot{p}_i = -\frac{\partial H}{\partial q_i} = \frac{\partial \lg \pi(q)}{\partial q_i} \qquad (4.3)$$

(The proof of $\dot{q}_i = \frac{\partial H}{\partial p_i}$ and $\dot{p}_i = -\frac{\partial H}{\partial q_i}$ is given in Section A.1).

The stochastic dynamics then builds upon this to follow the trajectories in phase space where, because the H is time-independent, they can avail themselves of the standard properties of a time-independent, separable Hamiltonian [1]; these standard properties imply

---

[1] These properties are: constant H, reversibility, and Liouville's theorem [46].

that the canonical distribution (the Gibbs distribution using the Hamiltonian) is invariant

with respect to any transformation that consists of following Hamiltonian dynamics for some

period of time [46], as shown in Figure 4.1.

Figure 4.1: Overview of the Traditional HMC Algorithm. This shows how the Hamiltonian evaluated at the initial phase space point, (q^o, p^o), is compared to the Hamiltionian evaluated at the final phase space point, (q', p'), in the Dynamic Transition Step using LeapFrog (Step 2).

The dynamics are implemented using the Leapfrog algorithm since, in practice,

we cannot follow the dynamics exactly (analytically) [46]. The *Dynamic Transitions* im-

plemented by the Leapfrog algorithm (described in [46, 4]) allow us to discretize these

equations using some non-zero time step. The systematic error in sampling introduced from

the discretization of time into units Δt in the Leapfrog algorithm in both the trajectory and

the energy, H, is removed by a Markov Chain, based on the stochastic transitions in the

Metropolis-Hastings, as shown in Algorithm 1 (please also refer to its analysis in Table 4.4).

These stochastic transitions allow points in phase space with different energies to be visited

and ensure the Markov Chain is ergodic (can visit all points of a given energy). HMC thus samples points in phase space by means of a Markov Chain in which stochastic and dynamic transitions alternate.

4.3 Data Driven HMC

4.3.1 Overview

We propose a physically driven framework for the HMC architecture by combining the more traditional energy-based Hamiltonian approach with data-driven proposals derived from video observations. The resulting approach is called the Data Driven HMC (DDHMC). We propose two different forms of the DDHMC: $DDHMC_{Motion}$, which uses motion-based data-driven proposals, and $DDHMC_{Shape}$, which uses shape-based data-driven proposals. Both variants of the DDHMC rely upon a data-driven component to make more informed proposals than the blind proposals generated within a Traditional HMC. These informed proposals, based on the likelihood of a particular track under a KDE or Gibbs estimator, are then used as the data-driven portion of the HMC to do an initial classification of the activities.

We assume we have similarity distributions using both the Motion and the Shape methods. We then convert these to probability density functions by casting them as a Gibbs or Kernel Density estimator. This results in a joint distribution, $\pi(\tau, f) = \pi(\tau|f)\pi(f) = \pi(f|\tau)\pi(\tau)$, where $\pi(f)$ is the probability density function for the Form or Shape pathway and $\pi(\tau)$ is the probability density function for the Motion pathway. Our goal is to sample this joint space, $\pi(\tau, f)$, and we employ our DDHMC variants to do exactly this since the

69

HMC has proved so successful in analyzing high-dimensional spaces in phase space. We expect the peaks to be highest in our joint distribution where both individual distributions exhibit higher values and we use the data-driven proposals to narrow in on those areas specifically. Because we expect the peaks in the joint distribution to correspond to areas where peaks of the motion and form distributions maximally overlap, we can use the DDHMC to sample from just the $\pi(\tau)$ or the $\pi(f)$ instead of the $\pi(\tau, f)$, as well. For example, in the $DDHMC_{Motion}$, we create a Hamiltonian that is a combination of Shape and HES, so we sample from $\pi(\tau, f)$ in this case; in the $DDHMC_{Shape}$, we sample from the distribution of motion similarities, $\pi(\tau)$.

Thus, both variants require a similarity distribution; for $DDHMC_{Shape}$, we use shape-based methods to compute similarities between the shapes obtained from a query track and all the database test tracks. For $DDHMC_{Motion}$, we use the physically significant Hamiltonian derived from tracks (please see Section 4.3.2) and Dynamic Time Warping (DTW) to do the same. Since we use DTW, in future work, we can potentially employ a lower bounding function to speed up the similarity search [36], especially when dealing with large databases; in fact, the lower bounds can be used in the proposal and confirmation searches separately to speed up the DDHMC since the DDHMC itself provides integration and not just search over a single component or pathway.

Our approach assumes that the video is segmented into objects and their motion is given; then, from the physical motion and location information of objects over time, we can use the abstract Hamiltonian framework to estimate the potential and kinetic energy of the system and derive an approximation to its Hamiltonian. Thus, starting with tracks for an object, we calculate the Hamiltonian for each object. The integration of a Hamiltonian

energy-based approach with an image-based data-driven proposal allows us to simultaneously search over both the motion energy space and image space in a concerted manner, unlike traditional HMC methodologies which only address a single space, thus reducing the enormity of the search space. We can then classify activities based on the likelihood of a particular track under a KDE or Gibbs estimator.

These are the Data-Driven proposals we use in our two DDHMC formulations: $DDHMC_{Motion}$ and $DDHMC_{Shape}$. For the first variant, $DDHMC_{Motion}$, we use proposals from the motion space and the same Dynamic Transitions as for the Traditional HMC, as seen in Algorithm 1. For the second variant, $DDHMC_{Shape}$, we use proposals from the shape space and then use Dynamic Evolution of the the Proposal and Acceptance Hamiltonians calculated from the tracks (explained in Section 4.4.2). Dynamic transitions, as explained in [46], refers to the Hamiltonian dynamics shown in (4.3) being applied for l time steps and then testing the candidacy of that point based on the change in total energy using a Metropolis-Hastings step. Dynamic Evolution is our innovation for the DDHMC in which we follow a trajectory for l time steps and then make a similar decision to accept or reject based on the *difference* in energy between the Proposal and Acceptance Hamiltonians using a Metropolis-Hastings step. Since the physically-significant Hamiltonian and the distribution estimator are common to both variants, we explain them in the next two sections before we detail the two DDHMC algorithms.

4.3.2 Derivation of Hamiltonian from video

In order to derive the Hamiltonian dynamics for systems studied in video, we follow the development in Section 2.2 and build upon equations (2.1) - (2.3). If the transformation

between the Cartesian and generalized coordinates is time-independent, then the Hamiltonian function also represents the total mechanical energy of the system:

$$H(q(t), p(t)) = T(p(t)) + U(q(t)) \tag{4.4}$$

In general, we compute (2.3), which does depend on time, but we can make the assumption (4.4) as a first approximation, as discussed in the Appendix. The procedure [25] for deriving the Hamiltonian is to first write out the Lagrangian, L, from equation (2.2) in generalized coordinates, expressing T and U in the normal manner for Lagrange's equation. Then, the generalized momenta are calculated by differentiating the Lagrangian with respect to the generalized velocity as:

$$p_i = \frac{\partial L}{\partial \dot{q}_i} \tag{4.5}$$

Now we can express the generalized velocities in terms of the momenta by simply inverting the result of (2.5) and using those generalized velocities in (2.3). We thus finally arrive at the equations of motion of Hamiltonian mechanics, known as **Hamilton's Equations of Motion**, which we derive from the Hamiltonian equivalent of the Euler-Lagrange equations:

$$\frac{\partial H}{\partial p_i} = \dot{q}_i, \frac{\partial H}{\partial q_i} = F_i - \dot{p}_i, \frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t} \tag{4.6}$$

where, for a free particle with no external forces, the F_i term goes to zero, leaving:

$$\frac{\partial H}{\partial p_i} = \dot{q}_i, \frac{\partial H}{\partial q_i} = -\dot{p}_i, \frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t} \tag{4.7}$$

This defines the dynamics on the system's phase-space, in which the q_i and p_i are regarded as functions of time [38, 57]. *The phase–space of a system consists of all possible values of the generalized coordinate variables q_i and the generalized momenta variables p_i.* If the Hamiltonian is time-independent, then phase space is 2-dimensional, (q,p); if the Hamiltonian is time-dependent, then phase space is 3-dimensional, (q,p,t) [62].

4.3.2.1 Application to Activity Recognition

Our approach is to segment the video into systems and sub-systems (e.g., whole body of a person or parts of the body) and, for each of those, get their tracks, from which we compute T and U, and use that to get the Hamiltonian, which can then be evaluated further and the results analyzed accordingly, as shown in Figure 4.2.

Tracks (x,y,t) → KE & PE → Lagrangian (L) → Action (S) → Hamiltonian (H) → Phase Space

Figure 4.2: Tracks to Hamiltonian to Phase Space: the phase space of a system consists of all possible values of the coordinates, which can be (q,p) or (q,p,t), for example; we may also look at modified phase plots of (H,t), (H,q,p), etc. This is similar to Figure 2.1.

We use the video to gain knowledge of the physics and use the physics to capture the motion energy of the system being observed via the abstract Hamiltonian framework. In order to compute the Hamiltonian, we use tracks from the video to compute the kinematic quantities that drop out of the Lagrangian formalism, thus giving a theoretical basis for examination of their energy from (x,y,t).

A system, in this sense, is defined according to the constraints of the video and the systems we are trying to identify. Thus, a system could be points on the joints of a person, points on the contour of a person, the center of mass of an object, etc. More generally, the

Hamiltonian can be used to characterize any system or sub-system. We thus use the tracks to compute the kinematic quantities that drop out of the Lagrangian formalism, thereby giving a theoretical basis for examination of their energy from the (x,y,t) tracks (further details are provided in the attached Appendix). In the end, we take the tracks and map them to 3-dimensional phase space (q,p,t) trajectories, τ, from which we form the appropriate Hamiltonians, $H(q,p,t) = H(\tau)$, as shown in Figure 4.3, where we see the tracks for three cars engaged in an activity being mapped to an (H,t) phase plot.
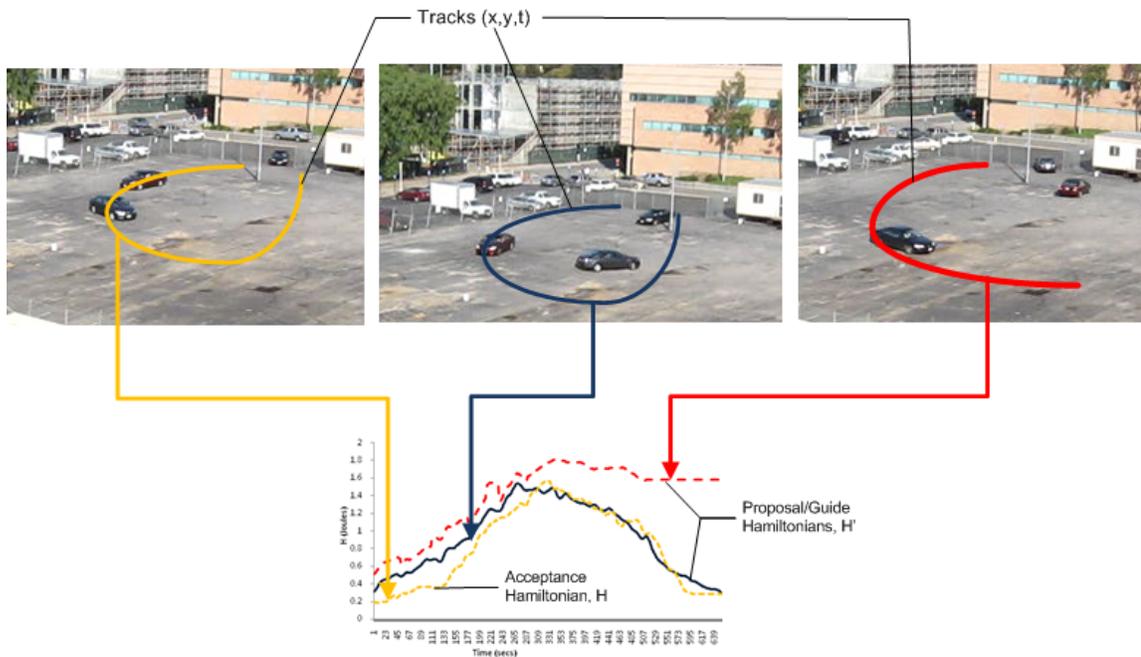
Figure 4.3: Map (x,y,t) tracks to an (H,t) phase plot. This is similar to Figure 2.3 and specifically shows the trajectories for two cars following. The first car, whose trajectory is labeled in orange, is the lead car and executes a U-turn and might represent the query which we could call the Acceptance Hamiltonian. The second car, trajectory in blue, follows it and also makes a U-turn, whereas the third car, whose trajectory is in red, follows it for a while and then turns away; both of these can become Proposal Hamiltonians representing the test clips from the database.

For example, if we track a person in video, we can compute these Hamiltonians for the centroid of the person (considering the person as an entire object) or consider all the

points on the contour of that person's silhouette, thus leading to a multi-dimensional time series (which can, for example, represent the gait of a person). Note that these Hamiltonians can be computed in either the image plane, yielding the Image Hamiltonian as used in this work, or in the 3D world, giving the Physical Hamiltonian, depending on the application domain and the nature of the tracks extracted. In either case, the rigorous Hamiltonian framework gives a highly abstract representation for a system and can yield the energy of the system under consideration (please see Appendix for details of the specific conditions). These Hamiltonians can then model a physically-significant energy that only depends on the image velocities. Since our representation uses the abstract Hamiltonian framework and is based upon the physical motion information of the system, garnered directly from the tracks, it is much more physically significant as compared to "energy-based" methods which take some property of the image, like pixel intensity, and square it or find its fourier energy [39, 47]. In the experiments shown in this work, we will use the Image Hamiltonian.

4.3.3 Proposal distribution formulation

In order to create a data-driven proposal, we need to first compute similarity scores between the query and test tracks, using either the motion energy or form/shape based methods, and then cast them as a distribution. In order to compute the motion energy similarity measures, we use Dynamic Time Warping (DTW) to compare the time series of the physically-significant Hamiltonian for each track.

Similarly, we use well-established methods in machine vision to calculate form or shape based features for our representation. Our construction provides flexibility on the image-based data-driven proposal side since different approaches in low-level feature

75

extraction can be employed easily within our framework and the appropriate image-based data-driven proposal features can be used. In general, we can follow the example of other data-driven MCMC approaches like [65]. For the present work, we use the shape of the feature points in the video, which can be the shape of a trajectory or the shape of our object. Distances can be computed between shapes, leading to a similarity matrix [68]. We can also use Histogram of Oriented Gradients (low-resolution experiments) [12], shape, trajectory-based descriptors, colour/texture, etc., since the integration is directly on the similarity scores.

We then cast the similarity scores from our data-driven method as a Gibbs proposal distribution since any distribution that is nowhere zero can be put in a canonical (Gibbs) distribution form [46, 20]. In order to estimate the distribution for the similarity scores, we use standard Kernel Density Estimation [8]:

$$
\begin{aligned}
K(D) &= \frac{1}{nh} \sum_{i=1}^{n} K_{eff}\left(\frac{D-d_i}{h}\right) \\
&\text{with } K_{eff}(D) = (2\pi)^{-\frac{1}{2}} e^{-\frac{D^2}{2}} \text{ and } d_i \in (D-h, D+h]
\end{aligned}
\tag{4.8}
$$

where D is the distance measure between two tracks and h is the bandwidth, which is set using kNN, as described in [8].

4.4 DDHMC variants

4.4.1 DDHMC with Shape-based Proposals: $DDHMC_{Shape}$

The $DDHMC_{Shape}$ relies upon shape-based proposals and we address its development within the context of a video database-retrieval. Suppose we have a query clip and a video database of clips and we would like to match the activity in the query clip with the

76

activity in each of the test video database clips. We assume that we have tracks for all the objects in each clip. We then apply our shape-based method to all the clips in the database and compute distance measures between each test clip and the query clip. This will then be cast as the distribution, following the formulation in Section 4.3.3, for our data-driven proposals in this case (Step 1 of the HMC algorithm).

For Step 2 of the HMC algorithm, we take each proposal clip and compute physically-significant Hamiltonians, as shown in Section 4.3.2, for the object in the query clip and the object in the proposal test clip. We compare the two Hamiltonians for l time steps using DTW and accept their difference with a certain probability, α. We keep track of the number of accepted and the number of rejected time steps via the $\frac{N_{rejected}}{N_{accepted}}$ ratio.

Finally, in Step 3 of the HMC algorithm, we use that $\frac{N_{rejected}}{N_{accepted}}$ ratio in a Metropolis-Hastings test to see if we should accept the proposal test clip or not.

4.4.2 Detailed Description of the $DDHMC_{Shape}$ Algorithm

Our full algorithm for the $DDHMC_{Shape}$ is outlined in Algorithm 2 but we describe it in detail here and also give an analysis of the algorithm in Table 4.6. In addition, please refer to Table 4.2 for an overview of the differences between the DDHMC and the Traditional HMC algorithms.

A diagrammatic representation of the overall evolution and eventual matching approach of the algorithm in trajectory space is shown in Figure 4.4. Figure 4.4(a) shows seven trajectories in trajectory space (represented as yellow circles).

Lines 3-11 of Algorithm 2 start off in Figure 4.4(b), where we enter Step 1 of the algorithm and find a proposal trajectory using the image-based data-driven methodology

77
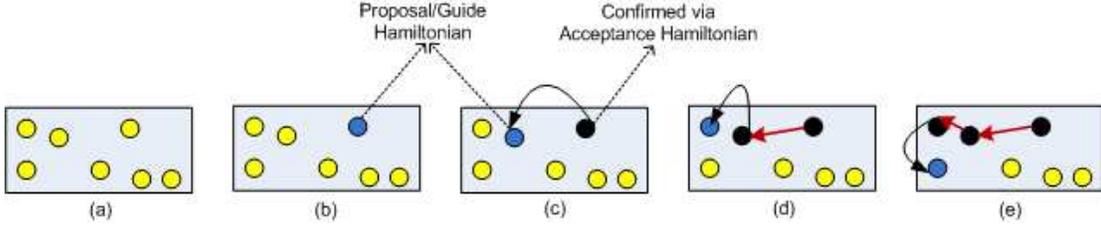
Figure 4.4: Proposal/Acceptance in Trajectory Space (a) shows seven trajectories in trajectory space (represented as yellow circles). Algorithm 2 starts off in (b) and chooses a Proposal Hamiltonian (lines 3-11). This Proposal Hamiltonian is then compared with the Acceptance Hamiltonian (lines 12-22) and, if it is accepted (lines 23-29), the algorithm continues with the loop by finding a new Proposal Hamiltonian in (c). In this way, the algorithm maneuvers through trajectory-space, only picking out those trajectories whose Proposal Hamiltonians are confirmed by the Acceptance Hamiltonian of the query clip, (c)-(e).

described earlier; the selected proposal trajectory is then used to create the Proposal/Guide Hamiltonian (the blue circle). As described variously [18, 46, 42], although the Traditional HMC uses a single Hamiltonian for both Acceptance and Guidance, the more general HMC formulation allows the use of a "Guide/Guidance" Hamiltonian, H', which does not have to be the same as the "Acceptance" Hamiltonian, H; this is the approach we use in $DDHMC_{Shape}$, as well.

Putting this in the context of matching a database query, we first use the shape-based method to compute distances between the query clip and all the test clips in the database. We cast these similarity measures as a distribution, as shown in Section 4.3.3, for our data-driven proposals in this case. In Step 1 of the algorithm (lines 3-11), we create a MCMC that checks these scores and picks a test clip that has a better match. This clip becomes our selected proposal and we compute its Hamiltonian to create the "Proposal/Guide" Hamiltonian. Similarly, we compute a Hamiltonian for the query clip, which becomes the "Acceptance" Hamiltonian. The Acceptance Hamiltonian decides the acceptance/rejection and is derived from our query/probe whereas the Proposal/Guide Hamiltonians, H', are

derived from the gallery/database and are our data-driven proposals.

This Proposal Hamiltonian is then compared with the Acceptance Hamiltonian in the Dynamic Evolution step (lines 12-22), shown in Figure 4.5, where we allow both the Guide and Acceptance Hamiltonians to evolve together using the real time-steps given from video by the frame rate (since we use real time, we don't need to artificially discretize time

Figure 4.5: Dynamic Evolution: here, we see the trajectory for the Proposal Hamiltonian and the trajectory for the Acceptance Hamiltonian. We also see two points in time along those trajectories at t^j and t^{j+1}. At t^j, for example, we compute the difference between the two Hamiltonians up to that point using DTW; this difference is δH. If the difference is less than zero or within probability α, we increment $N_{accepted}$; else, we increment $N_{rejected}$. Then, we repeat this analysis at t^{j+1} and continue in this manner for l timesteps.

into step-sizes Δt, as in the Traditional HMC). As we see in Figure 4.5, at each step (e.g., t^j), we calculate $\delta H = H - H'$, between the Acceptance and Proposal/Guide Hamiltonians; because we're examining the difference over the whole trajectory up to that step, we use Dynamic Time Warping (DTW) to determine the actual difference.

If this difference is, in general, close to 0, we conclude that the two clips match. This is because, as pointed out by [62], in phase space, if H is explicitly time-dependent, we can assert that no two orbits can pass through the same phase space point at the same time. Ideally, the difference should be exactly zero but, due to the discretization errors mentioned earlier, we allow it to be close to zero as determined by the Metropolis-Hastings step, instead.

So we let both the Proposal and Acceptance Hamiltonians evolve and, if their

difference is overall negligible, as reflected in the $\frac{N_{rejected}}{N_{accepted}}$ ratio, we conclude that they are the same. Thus, if $N_{rejected} > N_{accepted}$, the proposed trajectory is not accepted (since $\alpha \not> 1$); however, if the proposed trajectory is close to the acceptance proposal, as compared to α, we conclude they are the same. Force bias methods of [51, 44] are variations that justify our differential approach.

Finally, if the Proposal Hamiltonian is accepted (lines 23-29), the algorithm continues with the loop by finding a new Proposal Hamiltonian in Figure 4.4(c), which might also be accepted. In this way, the algorithm maneuvers through trajectory-space, only picking out (accepting) those trajectories whose Proposal Hamiltonians are confirmed by the Acceptance Hamiltonian of the query clip, Figure 4.4(c)-(e).

We therefore allow dynamic evolution for a specified time to account for the discretization errors referenced for the traditional HMC. In fact, although our system is based on the assumption of a time-dependent Hamiltonian, [3] points out that the time-dependence in H averages itself out up to negligible terms so any cross-over of points (i.e., any time when $\delta H = 0$) further indicates they are the same phase space trajectory. The stepsize in the Traditional HMC and the length, l, in both are also a way to represent "smoothing" in our system.

Our approach , in brief, is to:

- Create an Acceptance Hamiltonian , H, from the query (see above for details of H and H')

- Sample a Guide Hamiltonian , H', from the Gibbs distribution of the data-based approach as our data-driven proposal

80

Lines	Analysis of Algorithm 2
1	Initialize the Markov Chain of accepted trajectories, $\{\tau_i\}$, which we'll eventually return in Line 31
2	Start the main for loop
3-5	Start the Data-Driven Loop
6-7	Draw a proposal trajectory τ_i' from the data-based Gibbs distribution and draw a random number, α
8-11	Check if $D_{Shape}(\tau_i')$ is less than $D_{Shape}(\tau_{i-1})$ and accept τ_i' with probability α
12	Start the Dynamic Evolution step
13	Pick a random time j^o within the trajectory τ_i'
14	Let the system evolve for l steps
15	At each time step, compute the difference in Hamiltonians between the proposal trajectory, τ_i', and the acceptance/query trajectory, τ_q, using DTW from that time step to the beginning of each trajectory.
16-22	Draw a random number, α, and accept the difference with that probability; increment $N_{accepted}$ if accepted; otherwise, increment $N_{rejected}$.
23-24	Start the Final Metropolis-Hastings step and draw a random number, α
25-30	Compute the ratio, $\frac{N_{rejected}}{N_{accepted}}$, and accept the proposal trajectory, τ_i', with probability, α
31	Return the final Markov Chain of accepted trajectories, $\{\tau_i\}$

Table 4.6: Analysis of Algorithm 2

Algorithm 2 Data Driven HMC Algorithm - Shape-based Proposals

Outline of Data Driven Hamiltonian Monte Carlo which builds a Markov Chain over proposal trajectories, τ_i. τ_o is the initial trajectory from the gallery/database, τ_q is the trajectory of the query, $nsamples$ is the number of samples in the gallery/database, and $D_{Shape}(\tau_o, \tau_q)$ is the shape/form-based distance measure for the trajectory τ_o from τ_q. $H(\tau_q)$ is the Acceptance Hamiltonian and $H'(\tau_i)$ is the Proposal/Guide Hamiltonian and $H(\tau;t) = H(q(t), p(t))$ for the trajectory $\tau(q, p)$.

1: Initialize chain with τ_o
2: **for** $i = 1$ to $nsamples$ **do**
3:    // 1. Data-Driven: Get Proposal/Guide trajectory τ_i' from data-based Gibbs distribution
4:    flag $\leftarrow$ true
5:    **while** (flag) **do**
6:       draw $\tau_i' \sim e^{-K(D_{Shape}(\tau_i, \tau_q))}$
7:       draw $\alpha \sim \mathcal{U}[0, 1]$
8:       **if** $\alpha > min\left(1, \frac{D_{Shape}\left(\tau_i', \tau_q\right)}{D_{Shape}(\tau_{i-1}, \tau_q)}\right)$ **then**
9:          flag $\leftarrow$ false
10:       **end if**
11:    **end while**
12:    // 2. Dynamic Evolution – Time Dependent Step: sample a time-step uniformly from τ_i'
13:    $j^o \sim \mathcal{U}[\tau_i']$
14:    **for** $j = j^o$ to $j^o + l$ **do**
15:       $\delta H = \left\| H'\left(\tau_i'; t^j\right) - H(\tau_q; t^j) \right\|$ or $\delta H = DTW\left(H'\left(\tau_i'; t^j\right) - H(\tau_q; t^j)\right)$
16:       draw $\alpha \sim \mathcal{U}[0, 1]$
17:       **if** $\alpha < min\left(1, e^{-\delta H}\right)$ **then**
18:          $N_{accepted}++$
19:       **else**
20:          $N_{rejected}++$
21:       **end if**
22:    **end for**
23:    // 3. Final Metropolis-Hastings: accept data-driven proposal with certain probability
24:    draw $\alpha \sim \mathcal{U}[0, 1]$
25:    **if** $\alpha > min\left(1, \frac{N_{rejected}}{N_{accepted}}\right)$ **then**
26:       $\tau_i = \tau_i'$
27:    **else**
28:       $\tau_i = \tau_{i-1}$
29:    **end if**
30: **end for**
31: return $\{\tau_i\}_{i=0}^{nsamples}$

- Dynamically evolve, over real time, our Hamiltonians for a length, l

- At each step, evaluate a Metropolis-Hastings (MH) acceptance ratio for $\delta H = H' - H$

- Finally, grow the Markov Chain using another MH test on the ratio of $\frac{N_{rejected}}{N_{accepted}}$ proposals

An overview of the proposal generation is shown in Figure 4.6, where we see that the Shape-based proposal suggests a trajectory, τ, which is then analyzed within the physically-significant H to make the final decision for acceptance or rejection.

4.4.3 DDHMC with Motion-based Proposals: $DDHMC_{Motion}$

We can also reverse the DDHMC proposal and acceptance methods. We propose this second approach to the integration of form and motion energy information because, depending on the specific application or problem, either the form or the motion might outperform the other. Since our integration affords a hierarchical classification scheme in which the data-driven proposal does an initial, gross classification, we create both variants to take best advantage of whichever method is most suited to the finer granularity of classification; e.g., in image analysis, the motion-based proposals give a gross classification and the image-based confirmation via the traditional HMC architecture yields the final, finer classification in the integration afforded by the DDHMC framework. Thus, in Algorithm 3, we use the Hamiltonian analysis to generate the proposals whereas the shape or form method confirms the acceptance.

As can be seen there, the main difference from the $DDHMC_{Shape}$ is the use of motion energy based proposals in Step 1 (lines 3-11). We still use an Acceptance and

```
1:  Initialize chain with $\tau_0$
2:  for $i = 1$ to $nsamples$ do
3:      // 1. Data-Driven: Get Proposal/Guide trajectory $\tau_i'$ from data-based Gibbs distribu-
        tion
4:      flag $\leftarrow$ true
5:      while (flag) do
6:          draw $\tau_i' \sim e^{-K(D_{Shape}(\tau_i, \tau_q))}$
7:          draw $\alpha \sim \mathcal{U}[0, 1]$
8:          if $\alpha > min\left(1, \frac{D_{Shape}(\tau_i', \tau_q)}{D_{Shape}(\tau_{i-1}, \tau_q)}\right)$ then
9:              flag $\leftarrow$ false
10:         end if
11:     end while
12:     // 2. Dynamic Evolution – Time Dependent Step: sample a time-step uniformly from
        $\tau_i'$
13:     $j^o \sim \mathcal{U}[\tau_i']$
14:     for $j = j^o$ to $j^o + l$ do
15:         $\delta H = \left\| H'(\tau_i'; t^j) - H(\tau_q; t^j) \right\|$ or $\delta H = DTW\left(H'(\tau_i'; t^j) - H(\tau_q; t^j)\right)$
16:         draw $\alpha \sim \mathcal{U}[0, 1]$
17:         if $\alpha < min(1, e^{-\delta H})$ then
18:             $N_{accepted}$++
19:         else
20:             $N_{rejected}$++
21:         end if
22:     end for
23:     // 3. Final Metropolis-Hastings: accept data-driven proposal with certain probabil-
        ity
24:     draw $\alpha \sim \mathcal{U}[0, 1]$
25:     if $\alpha > min\left(1, \frac{N_{rejected}}{N_{accepted}}\right)$ then
26:         $\tau_i = \tau_i'$
27:     else
28:         $\tau_i = \tau_{i-1}$
29:     end if
30: end for
31: return $\{\tau_i\}_{i=0}^{nsamples}$
```

Gibbs Distribution (Geman & Geman)

Shape Space → Shape Proposal → Motion Space → Trajectories

Rejected
Accepted

δH

t^i, t^{i+1}, t^{i+2}, t^{i+3}, t^{i+4}

Acceptance Hamiltonian, H

Proposal/Guide Hamiltonian, H'

Figure 4.6: Shape-Proposal Generation Overview: here we see that the Shape-based proposal suggests a trajectory, τ, which is then analyzed within the physically-significant H to make the final decision for acceptance or rejection

Lines	Analysis of Algorithm 3
1	Initialize the Markov Chain of accepted trajectories, $\{\tau_i\}$, which we'll eventually return in Line 31
2	Start the main for loop
3-5	Start the Data-Driven Loop
6-7	Draw a proposal trajectory $\tau_i^{'}$ from the data-based Gibbs distribution and draw a random number, α
8-11	Check if $D_{HES}(\tau_i^{'})$ is less than $D_{HES}(\tau_{i-1})$ and accept $\tau_i^{'}$ with probability α
12-14	Initialize the q and p using the data-driven proposal trajectory $\tau_i^{'}$
15-21	Let the system evolve dynamically (using the Leapfrog algorithm) from the initial point in phase space, (q^o, p^o), to the final point in phase space, $(q^{'}, p^{'})$.
22-24	Start the Final Metropolis-Hastings step and draw a random number, α, and compute the difference between the current Hamiltonian for the final point, $(q^{'}, p^{'})$, and the Hamiltonian for the previously accepted point, (q_{i-1}, p_{i-1}).
25-30	If the difference is 0 or less, always accept the new phase space point, $(p^{'}, q^{'})$, since this means the two trajectories intersected in phase space and so they have the same H; otherwise, accept it with probability α
31	Return the final Markov Chain of accepted trajectories, $\{\tau_i\}$

Table 4.8: Analysis of Algorithm 3

Algorithm 3 Data Driven HMC Algorithm – Motion-based Proposals

Outline of Data Driven Hamiltonian Monte Carlo which builds a Markov Chain over proposal trajectories, τ_i. τ_o is the initial trajectory from the gallery/database, τ_q is the trajectory of the query, *nsamples* is the number of samples in the gallery/database, and $D_{HES}(\tau_o, \tau_q)$ is the motion energy distance measure for the trajectory τ_o from τ_q and $D_{Shape}(\tau_o, \tau_q)$ is its shape/form-based distance. $H(\tau_q)$ is the Acceptance Hamiltonian and $H'(\tau_i)$ is the Proposal/Guide Hamiltonian and $H(\tau;t) = H(q(t), p(t))$ for the trajectory $\tau(q, p)$.

1: Initialize chain with τ_o
2: **for** $i = 1$ to *nsamples* **do**
3:    // 1. Data-Driven: Get Proposal/Guide trajectory τ_i' from data-based Gibbs distribution
4:    flag $\leftarrow$ true
5:    **while** (flag) **do**
6:       draw $\tau_i' \sim e^{-K(D_{HES}(\tau_i, \tau_q))}$
7:       draw $\alpha \sim \mathcal{U}[0, 1]$
8:       **if** $\alpha > min\left(1, \dfrac{D_{HES}(\tau_i', \tau_q)}{D_{HES}(\tau_{i-1}, \tau_q)}\right)$ **then**
9:          flag $\leftarrow$ false
10:       **end if**
11:    **end while**
12:    // Initialize q and p using the Data-Driven proposal trajectory τ_i'
13:    $q^o = D_{Shape}(\tau_i', \tau_q)$
14:    $p^o = D_{HES}(\tau_i', \tau_q)$ // can also be: draw $p^o \sim \mathcal{N}(0, 1)$
15:    // 2. Do Perturbation using Dynamic Transition via Leapfrog
16:    **for** j−1 to l **do**
17:       $p^{j-\frac{1}{2}} = p^{j-1} - \frac{\Delta t}{2} \bullet \nabla U(q^{j-1})$
18:       $q^j = q^{j-1} - \Delta t \bullet p^{j-\frac{1}{2}}$
19:       $p^j = p^{j-\frac{1}{2}} - \frac{\Delta t}{2} \bullet \nabla U(q^j)$
20:    **end for**
21:    $(q', p') = (q^l, p^l)$
22:    // 3. Final Metropolis-Hastings: accept data-driven proposal with certain probability
23:    draw $\alpha \sim \mathcal{U}[0, 1]$
24:    $\delta H = H'(q', p') - H(q_{i-1}, p_{i-1})$
25:    **if** $\alpha < min(1, e^{-\delta H})$ **then**
26:       $(q_i, p_i) = (q', p')$
27:    **else**
28:       $(q_i, p_i) = (q_{i-1}, p_{i-1})$
29:    **end if**
30: **end for**
31: return $\{\tau_i(q_i)\}_{i=0}^{nsamples}$

a Proposal/Guide Hamiltonian but the Dynamic Evolution of $DDHMC_{Shape}$ is replaced with a Perturbation (lines 15-21) in the $DDHMC_{Motion}$. For this step, we form a pseudo-Hamiltonian, exactly the same as in Traditional HMC, except we use the distance measures for the shape and motion as the generalized coordinates and momentum, respectively (lines 13-14), to create the Proposal Hamiltonian.

This Proposal Hamiltonian is then subjected to a perturbation via Dynamic Transitions using Leapfrog. The perturbation is done in phase space (which is more abstract than the form-based space). The reason for the perturbation is because we assume the Shape/Form method is not perfect and the perturbation, just like the Dynamic Transitions in the Traditional HMC, accounts for such errors. The Acceptance Hamiltonian, meanwhile, is computed for the previous phase space point, (q_{i-1}, p_{i-1}), that was accepted.

In Step 3, a normal HMC Metropolis-Hastings is used on the difference between the Acceptance and Proposal/Guide Hamiltonians. We finally accept the proposed trajectory if $\delta H \leq 0$ because it penetrates the Acceptance Hamiltonian's trajectory in phase space then (and so, we conclude the Guide Hamiltonian's trajectory is the same); but if $\delta H > 0$, we only accept with probability α.

An overview of the proposal generation is shown in Figure 4.7, where we see that the motion-based proposal suggests an artificial momentum, p, and the shape-based method is used within the HMC framework to get the artificial position coordinate, q; finally, both the q and the p are used to create the Hamiltonian, $H(q, p)$, which is then analyzed via the HMC framework to make the final acceptance decision.

In future versions of the DDHMC, we intend to generalize our approach beyond just the proposal/guide and acceptance Hamiltonians. In addition, we can also extend future

1: Initialize chain with τ_o
2: **for** $i = 1$ to $nsamples$ **do**
3:   // 1. Data-Driven: Get Proposal/Guide trajectory $\tau_i^{'}$ from data-based Gibbs distribution
4:   flag $\leftarrow$ true
5:   **while** (flag) **do**
6:     draw $\tau_i^{'} \sim e^{-K(D_{HES}(\tau_i,\tau_q))}$
7:     draw $\alpha \sim \mathcal{U}[0,1]$
8:     **if** $\alpha > min\left(1, \frac{D_{HES}\left(\tau_i^{'},\tau_q\right)}{D_{HES}\left(\tau_{i-1},\tau_q\right)}\right)$ **then**
9:       flag $\leftarrow$ false
10:     **end if**
11:   **end while**
12:   // Initialize q and p using the Data-Driven proposal trajectory $\tau_i^{'}$
13:   $q^o = D_{Shape}(\tau_i^{'},\tau_q)$
14:   $p^o = D_{HES}(\tau_i^{'},\tau_q)$ // can also be: draw $p^o \sim \mathcal{N}(0,1)$
15:   // 2. Do Perturbation using Dynamic Transition via Leapfrog
16:   **for** $j=1$ to l **do**
17:     $p^{j-\frac{1}{2}} = p^{j-1} - \frac{\Delta t}{2} \bullet \nabla U(q^{j-1})$
18:     $q^j = q^{j-1} - \Delta t \bullet p^{j-\frac{1}{2}}$
19:     $p^j = p^{j-\frac{1}{2}} - \frac{\Delta t}{2} \bullet \nabla U(q^j)$
20:   **end for**
21:   $(q^{'},p^{'}) = (q^l,p^l)$
22:   // 3. Final Metropolis-Hastings: accept data-driven proposal with certain probability
23:   draw $\alpha \sim \mathcal{U}[0,1]$
24:   $\delta H = H^{'}(q^{'},p^{'}) - H(q_{i-1},p_{i-1})$
25:   **if** $\alpha < min\left(1,e^{-\delta H}\right)$ **then**
26:     $(q_i,p_i) = (q^{'},p^{'})$
27:   **else**
28:     $(q_i,p_i) = (q_{i-1},p_{i-1})$
29:   **end if**
30: **end for**
31: return $\{\tau_i(q_i)\}_{i=0}^{nsamples}$

Figure 4.7: Motion-Proposal Generation Overview: here we see that the motion-based proposal suggests an artificial momentum, p, and the shape-based method is used within the HMC framework to get the artificial position coordinate, q; finally, both the q and the p are used to create the Hamiltonian, $H(q,p)$, which is then analyzed via the HMC framework to make the final acceptance decision

versions to compute an H_{eff} using the Action [18, 32] or the Action directly instead of H [17, 5, 11, 48].

4.5 Experiments

The DDHMC has several innovations that can be highlighted through experiments. Experiments on the Weizmann dataset demonstrate the reduction the search space by the data-driven portion as the final results are better than the original, as shown in Figure 4.8. For all of these experiments, tracking and basic object-detection was already available [33] and we utilized these (x,y,t) tracks to compute the Lagrangian and Hamiltonian, following our development in Section 4.3.2.1. For want of space, details are presented in the attached Appendix.

We show how the integration afforded by DDHMC helps reduce the search space. In this case, we demonstrate on the Weizmann dataset. The Weizmann dataset (`http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html`) consists of a database of 90 low-resolution (180 x 144, deinterlaced 50 fps) video sequences showing nine different people, each performing 10 natural actions. We analyze these using both shape methods [68] (as discussed in Section 4.3.3), as well as via the Hamiltonian. Using both procedures, we see the resulting similarity matrices in Figure 4.8 (a) and (b), respectively.

Finally, in Figure 4.8 (c) and (d), we see the result of integrating via DDHMC. As can be seen in the matrices, the DDHMC approach significantly reduces the search space of either the motion or shape approaches by themselves.
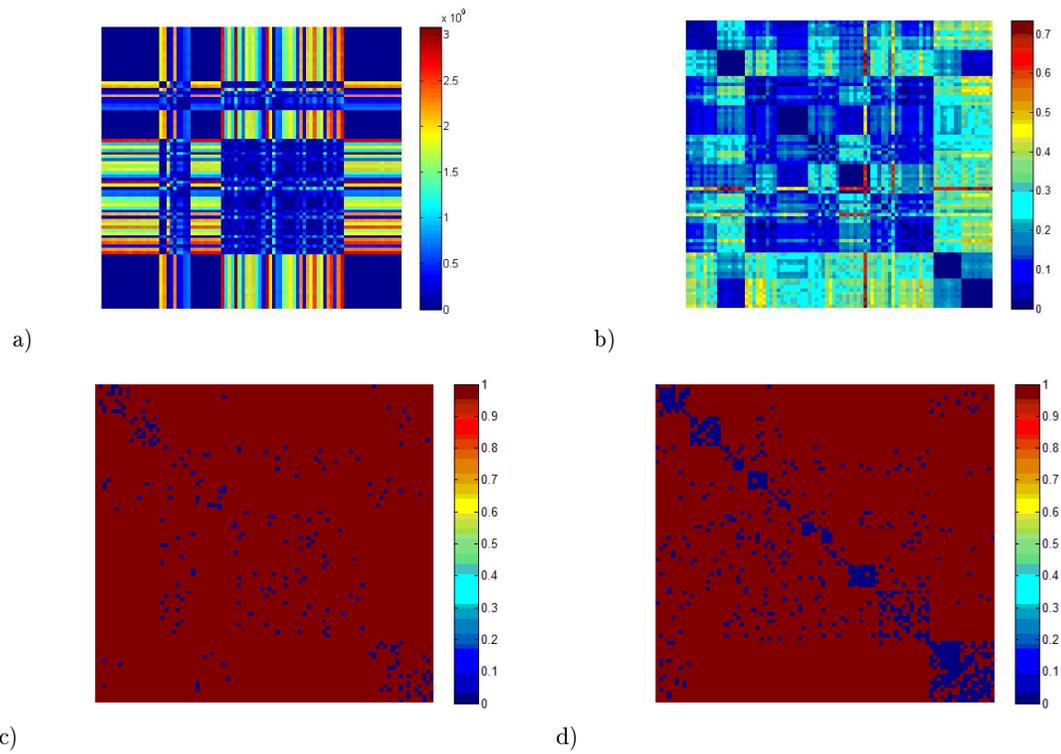
Figure 4.8: Similarity matrices using the Weizmann dataset for a) HES only, b) Shape Methods only, c) Integration using $DDHMC_{Motion}$, and d) Integration using $DDHMC_{Shape}$. The rows and columns represent activities by people and are organized according to activity. The plots show the clarification of matches using the finer granularity of either shape (in (c)) or motion in (d)).

Chapter 5

Conclusion

In this work, we propose several novel contributions to address the problem of integration of motion and form in order to do activity recognition in video. We develop a computational equivalent for the motion energy pathway of the neurobiological model based upon a fundamental physics formulation. Using the rigorous Hamiltonian framework, we propose Hamiltonian Energy Signatures (HES) as an abstract feature for detection of motion energy and activity recognition. The HES is view-invariant and can easily be generalized across different application domains and even be applied to coupled systems, like cars chasing each other, exchanges, or interactions between sparse objects, and other systems without requiring separate heuristics for each.

In addition, we extend our physical development to create a new spatio-temporal gait representation, called the Gait Action Image. We then create various statistical Integration mechanisms to combine both the motion and form pathways of the neural model.

The framework we present in this work provides a structured approach to motion recognition using the motion analysis neurobiological models within a single, unifying ar-

chitecture that only requires tracks for the Motion Energy Pathway. We believe it is the first computational equivalent for the integration of the two pathways of motion recognition that is applied to real world datasets. Our formulation takes an altogether novel approach whereby we attempt to create a theoretical framework inspired by the biological model and rooted in physics to gain insight into the problem of motion recognition in machine vision.

Finally, we develop yet another Integration variant motivated by the DDMCMC but that builds upon a physically-significant Hamiltonian Monte Carlo, which we call the Data Driven Hamiltonian Monte Carlo (DDHMC). The framework and architecture we presented in this work also provides a structured approach to activity recognition using the DDHMC. In addition, we develop two variants of the DDHMC that can be applied specifically to activity recognition, one using form-based proposals and another using motion-based proposals.

5.1   Future Work

We see much room for future research for our work. In particular, we see a path to create a generalized DDHMC which can be applied to many problems beyond motion recognition. In addition, we can apply our various DDHMC variants to problems beyond activity recognition, as well. A specific application we're exploring is applying the DDHMC framework to optimize the lasing temperature for a material under a magnetic field generated by pylons; in this case, the configuration space of the pylons can serve as the data-driven component and the H_{EM} (electromagnetic Hamiltonian) can serve as the physical Hamiltonian. In addition, we can expand our approach to compute an H_{eff} using the Action or the

Action directly instead of H.

We also intend to work on improving robustness to noise in low-level features, integration of different features for the statistical Integration framework, and efficient database search mechanisms to help supplement the integration, including looking at ways to set lower bounds on the individual pathways for grossly different classes of activities and objects. In addition, we can invert the two pathways' biasing in the Integration module; finally, we can use shape or learning algorithms to determine mass and potentials.

Finally, we see much potential for future HES development; e.g., using Phase-Space Trajectories (including preservation of phase-space volumes) and Poisson Brackets directly. We also intend to address robustness fo our high-level approach to low-level errors in the tracks; techniques for addressing this include potentially creating a Stochastic HES. We also recognize the need to investigate the mathematical properties of the DDHMC thoroughly, especially in regards to establishing its convergence. We discuss details of drawbacks, technical points, and further work in the Discussion section of the attached Appendix. Also, all code will be available online.

Bibliography

[1] Edward H. Adelson and James R. Bergen. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am. A*, 2:284–299, 1985.

[2] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73:428–440, 1999.

[3] E. Akhmatskaya and S. Reich. The targeted shadowing hybrid monte carlo (tshmc) method. In *New Algorithms for Macromolecular Simulation*. Springer-Verlag, 2006.

[4] M. Alfaki. Improving efficiency in parameter estimation using the hamiltonian monte carlo algorithm. Master's thesis, University of Bergen, 2008.

[5] G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky, and K. G. Wilson. Langevin simulations of lattice field theories. *Phys. Rev. D*, 10:2736–2747, 1985.

[6] Diane M Beck and Sabine Kastner. Top-down and bottom-up mechanisms in biasing competition in the human brain. *Vision Research*, 2008.

[7] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *PAMI*, 23:257–267, 2001.

[8] Z. I. Botev, J. F. Grotowski, and D. P. Kroese. Kernel density estimation via diffusion. *Annals of Statistics*, 2009.

[9] Andres Bruhn, Joachim Weickert, and Christoph Schnorr. Lucas/kanade meets horn/schunck: combining local and global optic flow methods. pages 211–231. Int. J. Comput. Vision, 2005.

[10] Kiam Choo and David Fleet. Tracking people using hybrid monte carlo. *ICCV*, pages 321–328, 2001.

[11] M. Creutz, L. Jacobs, and C. Rebbi. Monte carlo computations in lattice gauge theories. *Physics Reports*, 95:201–282, 1983.

[12] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *In CVPR*, 2005.

[13] Gustavo Deco and Tai Sing Lee. The role of early visual cortex in visual integration: a neural model of recurrent interaction. *European Journal of Neuroscience*, pages 1–12, 2004.

[14] Gustavo Deco and Edmund Rolls. Neurodynamics of biased competition and cooperation for attention: A model with spiking neurons. *J Neurophysiol*, pages 295–313, 2005.

[15] R. Desimone and J. Duncan. Neural mechanisms of selective visual attention. *Annu. Rev. Neurosci.*, 18:193–222, 1995.

[16] P Dollar, V Rabaud, G Cottrell, and S Belongie. Behavior recognition via sparse spatio-temporal feature. VS-PETS, 2005.

[17] Simon Duane. Stochastic quantization versus the microcanonical ensemble: Getting the best of both worlds. *Nuclear Physics B*, pages 652–662, 1985.

[18] Simon Duane, A.D. Kennedy, B.J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195:216–222, 1987.

[19] B. Efron and R. J. Tibshiriani. *An Introduction to the Bootstrap, Monographs on Statistics and Applied Probability 57*. Chapman & Hall, 1993.

[20] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *PAMI*, 6:721–741, 1984.

[21] Martin A Giese. Neural model for the recognition of biological motion. In *Dynamische Perzeption 2*, pages 105–110. Infix Verlag, 2000.

[22] Martin A. Giese. Neural model for biological movement recognition. In *Optic Flow and Beyond*, pages 443–470. Kluwer Academic Publihers, 2004.

[23] Martin A Giese and Tomaso Poggio. Neural mechanisms for the recognition of biological movements and action. *Nature Reviews Neuroscience*, 4:179–192, 2003.

[24] Nigel H Goddard. *The Perception of Articulated Motion: Recognizing Moving Light Displays*. PhD thesis, University of Rochester, 1992.

[25] H. Goldstein. *Classical Mechanics*. Addison-Wesley, 1980.

[26] L Gorelick, M Blank, E Shechtman, M Irani, and R Basri. Actions as space-time shapes. *PAMI*, 29(12):2247–2253, 2007.

[27] J. Han and B. Bhanu. Individual recognition using gait energy image. *Workshop on Multimodal User Authentication (MMUA 2003)*, pages 181–188, December 2003.

[28] Ju Han and Bir Bhanu. Individual recognition using gait energy image. *PAMI*, 28:316–322, 2006.

[29] Min Hu, Saad Ali, and Mubarak Shah. Detecting global motion patterns in complex videos. In *ICPR*, 2008.

[30] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man and Cybernetics*, 34:334–352, 2004.

[31] H Jhuang, T Serre, L Wolf, and T Poggio. A biologically inspired system for action recognition. ICCV, 2007.

[32] H. Jirari, H. Kröger, C.Q. Huang, J.Q. Jiang, X.Q. Luo, and K.J.M. Moriarty. Monte carlo hamiltonian. *Nuclear Physics Proc Suppl*, 83, 1999.

[33] A. Kale, A.N Rajagopalan, Sundaresan.A., N. Cuntoor, A. Roy-Chowdhury, V. Krueger, and R. Chellappa. Identification of humans using gait. Sept. 2004.

[34] Eric Kandel, James Schwartz, and Thomas Jessell. *Principles of Neural Science*. McGraw-Hill Medical, USA, 4th edition, 2000.

[35] S Kastner and L Ungerleider. The neural basis of biased competition in human visual cortex. *Neuropsychologia*, pages 1263–1276, 2001.

[36] Eamonn Keogh. Exact indexing of dynamic time warping. *VLDB*, pages 406–417, 2002.

[37] Greg Kochanski. Confidence intervals and hypothesis testing, 02 2005.

[38] L D Landau and E M Lifshitz. *Course of Theoretical Physics: Mechanics*. Pergamon Press, Moscow,, 3rd edition edition, 1976.

[39] Yann LeCun, Sumit Chopra, Marc Aurelio Ranzato, and Fu-Jie Huang. Energy-based models in document recognition and computer vision. Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR), 2007.

[40] J.J. Little and Boyd.J.E. Recognizing people by their gait : The shape of motion. *Journal of Computer Vision Research*, 1(2), 1998.

[41] Jingen Liu, Saad Ali, and Mubarak Shah. Recognizing human actions using multiple features. In *ICPR*. CVPR, 2008.

[42] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2004.

[43] Zongyi Liu and Sudeep Sarkar. Simplest representation yet for gait recognition: Averaged silhoutte. In *ICPR*, 2004.

[44] M. Mezei. Distance-scaled force biased monte carlo simulation for solutions containing a strongly interacting solute. *Mol. Phys.*, 5:405–408, 1991.

[45] V. Navalpakkam and L. Itti. An integrated model of top-down and bottom-up attention for optimal object detection. pages 2049–2056. Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[46] Radford M. Neal. Probabilistic inference using markov chain monte carlo methods. Technical report, University of Toronto, 1993.

[47] R.J. Peters and L. Itti. Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention. CVPR, 2007.

[48] J. Polonyi and H.W. Wyld. Microcanonical simulation of fermionic systems. *Physical Review Letters*, 51:2257–2260, 1983.

[49] Eunice Poon and David Fleet. Hybrid monte carlo filtering: Edge-based people tracking. In *WMVC*, 2002.

[50] M Ranzato, F Huang, Y Boureau, and Y LeCun. Unsupervised learning of invariant feature hierarchies with application to object recognition. CVPR, 2007.

[51] M. Rao, C.S. Pangali, and B.J. Berne. On the force bias monte carlo simulation of water: Methodology, optimization and comparison with molecular dynamics. *Mol. Phys.*, 37:1779, 1979.

[52] Chelazzi L. Reynolds, J. H. and R. Desimone. Competitive mechanisms subserve attention in macaque areas v2 and v4. *J. Neurosci.*, 19:1736–1753, 1999.

[53] Mikel Rodriguez, Javed Ahmed, and Mubarak Shah. A spatio-temporal maximum average correlation height filter for action recognition. *CVPR*, 2008.

[54] Claudio Castellanos Sanchez, Bernard Girau, and Frederic Alexandre. A connectionist approach for visual perception of motion. In *Brain Inspired Cognitive Systems*. 2004.

[55] M E Sereno. *Neural Computation of Pattern Motion*. Cambridge MIT Press, 1993.

[56] T Serre, L Wolf, and T Poggio. Object recognition with features inspired by visual cortex. CVPR, 2005.

[57] R Shankar. *Principles of Quantum Mechanics*. Springer, 2nd edition edition, 1994.

[58] C. Siagian and L. Itti. Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. IROS, 2007.

[59] Rodrigo Sigala, Thomas Serre, Tomaso Poggio, and Martin Giese. Learning features of intermediate complexity for the recognition of biological motion. In *Artificial Neural Networks: Biological Inspirations, ICANN 2005*. Springer Berlin, 2005.

[60] E P Simoncelli and D J Heeger. A model of neuronal responses in visual area mt. *Vision Research*, 38:743–761, 1998.

[61] Cristian Sminchisescu and Bill Triggs. Hyperdynamics importance sampling. *ECCV*, pages 769–783, 2002.

[62] John R. Taylor. *Classical Mechanics*. University Science Books, 2005.

[63] V. Tomar. Atomistic simulations of strain rate dependent deformation behavior at continuum timescales. In *MRS Symposium Proceedings*, 2007.

[64] A. M. Treisman and G. Gelade. A feature-integration theory of attention. *Cogn. Psychol.*, 12:97–136, 1980.

[65] Z Tu, S Zhu, and H Shum. Image segmentation by data driven markov chain monte carlo. ICCV, 2001.

[66] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 2008.

[67] A. Veeraraghavan, A. Roy-Chowdhury, and R. Chellappa. Role of shape and kinematics in human movement analysis. *CVPR*, 2004.

[68] Ashok Veeraraghavan, Amit Roy-Chowdhury, and Rama Chellappa. Matching shape sequences in video with applications in human motion analysis. *PAMI*, pages 1896–1909, 2005.

[69] Li Yang and Marwan Jabri. Sparse visual models for biologically inspired sensorimotor control. pages 131–138. Proceedings Third International Workshop on Epigenetic Robotics, 2003.

[70] Alan Yuille and Daniel Kersten. Vision as bayesian inference: analysis by synthesis. *TRENDS in Cognitive Sciences*, 10(7):301–308, 2006.

[71] S.C. Zhu, R. Zhang, and Z.W. Tu. Integrating top-down/bottom-up for object recognition by ddmcmc. *CVPR*, 2000.

[72] Enrico Zio and Francesco Di Maio. Bootstrap and order statistics for quantifying thermal-hydraulic code uncertainties in the estimation of safety margins. *Science and Technology of Nuclear Installations*, 2008.

[73] A. M. Zoubir and B. Boashash. The bootstrap and its application in signal processing. *IEEE Signal Processing Magazine*, 15:56–76, 1998.

Appendix A

Hamilton's Variational Principle

Hamilton's Variational Principle states that the integral, S, taken along a path of the possible motion of a physical system, is a minimum (technically, an extremum [38]) when evaluated along the actual path of motion. This variation can be expressed as:

$$\delta S = \delta \int_{t_1}^{t_2} L(q, \dot{q}, t) dt = 0 \qquad (A.1)$$

where δ is an operation that represents a variation of any system parameter by an infinitesimal amount away from the value taken by that parameter when (A.1) is an extremum. If we express L in terms of generalized coordinates, $q = q(t)$, then the change in S when q is replaced by $q + \delta q$ is arrived at by requiring that the *first variation* be zero [38] to yield, after integration by parts:

$$\delta S = \left[\frac{\partial L}{\partial \dot{q}} \delta q \right]_{t_1}^{t_2} + \int_{t_1}^{t_2} \left(\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} \right) \delta q dt = 0 \qquad (A.2)$$

This can only be true if the integrand is zero identically, which gives rise to the

so-called **Euler-Lagrange** equations of the Lagrangian formalism:

$$\frac{\partial L}{\partial q} - \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} = 0 \tag{A.3}$$

The Hamiltonian formalism is related to the Lagrangian formalism by the *Legendre transformation*, from generalized coordinates and velocities $(q, \dot{q})$ to generalized coordinates and momenta (q, p), using the $\dot{q}_i$, the time derivative of the generalized coordinates. Thus, the **Hamiltonian function** is usually stated most compactly, in generalized coordinates, as [38]:

$$H = \sum_i p_i \dot{q}_i - L \tag{A.4}$$

where H is the Hamiltonian, p_i are the generalized momentum, and $\dot{q}_i$ are the time derivative of the generalized coordinates, as in (2.3).

A.1 Deriving the Hamiltonian

The procedure for deriving the Hamiltonian [25] is to first write out the Lagrangian, L, from equation (2.2) in generalized coordinates, expressing T and U in the normal manner for Lagrange's equation. Then, the generalized momenta are calculated by differentiating the Lagrangian with respect to the generalized velocity:

$$p_i = \frac{\partial L}{\partial \dot{q}_i} \tag{A.5}$$

Now we can express the generalized velocities in terms of the momenta by simply inverting the result of (A.5) and using those generalized velocities in (A.4). Finally, we derive

Hamilton's Equations of Motion from the Hamiltonian equivalent of the Euler-Lagrange equations:

$$\frac{\partial H}{\partial p_i} = \dot{q}_i, \frac{\partial H}{\partial q_i} = F_i - \dot{p}_i, \frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t} \tag{A.6}$$

where, for a free particle with no external forces, the F_i term goes to zero, leaving:

$$\frac{\partial H}{\partial p_i} = \dot{q}_i, \frac{\partial H}{\partial q_i} = -\dot{p}_i, \frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t} \tag{A.7}$$

The first two relations give *2n* first-order differential equations and are called Hamilton's canonical equations of motion. This effectively results in expressing 1st-order constraints on a *2n*-dimensional Phase Space, whereas the Lagrangian method expresses 2nd-order differential constraints on an n-dimensional Coordinate Space.

Furthermore, if the total energy is conserved then the work, W, done on the particle had to have been entirely converted to potential energy, U. This implies that U is solely a function of the spatial coordinates (x,y,z); equivalently, U can be thought of as purely a function of the generalized configuration coordinates, q_i. Rarely, U is also a function of $\dot{q}_i$, making for a velocity-dependent potential, but is still independent of the time t. Noether's theorem, in fact, guarantees that any conserved quantity (e.g., energy) corresponds to a symmetry: thus, the system can then be thought of as being independent with respect to some variable or coordinate. In this case, Noether's theorem implies the independence of the Lagrangian with respect to time, as long as energy is conserved in this process.