Interactive Control of Biomechanical Animation

Contribution to the GI Workshop:

Visualisierung – Rolle von Interaktivität und Echtzeit

GMD, Sankt Augustin, 2.-3. Juni 1992

T. Ertl, H. Ruder, K. Gruber, M. Günther, F. Hospach,
T. Krebs, J. Subke, K. Widmayer

Lehr- und Forschungsbereich Theoretische Astrophysik der Universität Tübingen

Abstract

Physical based animation can be generated by performing a complete dynamical simulation of multi-body systems. This leads to a complex system of differential equations which has to be solved incorporating biomechanical results for the physics of impacts. Motion control is achieved by interactively modifying the internal torques. Realtime response requires the distribution of the workload of the computation between a high-speed compute server and the graphics workstation by means of a remote procedure call mechanism.

1. Introduction

In the first phase of computer animation the traditional techniques of animation were brought to the computer resulting in computer animated films where the keyframes were linked by image-based and parametric interpolation. Especially when trying to compute aesthetic human movement it soon became obvious that a more realistic computer animation has to take into account the basic physical properties of the objects and the fundamental physical principles that govern their movement. In algorithmic animation the evolution of the state of a system of objects is not determined by interpolation. but by physical laws given either as algebraic formulae in the simple case or more complicated as set of coupled nonlinear differential equations. The most general approach for generating physically correct animation sequences is to perform a complete dynamical simulation of the given model taking into account all external and internal forces and torques. However, even rigid objects with only a few degrees of freedom or very simple elastic models require supercomputer performance and the design of a desired motion by controling the internal torques is a tedious highly interactive process.

2. The motion of a rigid body

Although there are several computer animation techniques involving interacting point masses, most of the objects of interest are rigid bodies. A rigid body is defined by its total mass and its tensor of inertia. Its position and orientation in space is determined by three cartesian coordinates for the position of the center of mass and by three Euler angles for the orientation of a body-fixed coordinate system. The equations of motion decouple into the Newton equation for the motion of the center of mass under the influence of external forces and Euler equations for angular velocities of the rotation relative to the center of mass described by the body-fixed axes under the influence of external torques. Those equations can be easily derived and integrated resulting in the physically correct motion of the rigid body.

Nevertheless, even for the simple case of a falling rod where the equations of motion can be solved analytically, the correct treatment of the impact of the falling rod with the ground is a non-trivial problem. Since there is only a very limited number of interesting animations involving non-interacting rigid bodies, it is worthwhile to investigate the physical aspects of an impact. Two idealized cases, the totally elastic and the totally inelastic behaviour will be demonstrated in animation sequences. For a more realistic model the reaction force of the ground which depends on the amount of deformation and on the instantaneous velocity has to be taken into account. The effect of the free parameters of a ground reaction force ansatz can visualized by interactively modifying them during the animation.

3. The mechanics of multi-linked models

For the modelling of human beings or animals with legs and arms multi-linked systems of extended bodies connected by joints are necessary. Developing a satisfactory model is by no means a trivial problem. The joints and their constraints must be correctly described as well as the mechanical properties of the body segments. The general procedure will be shown for the plane two-linked rigid model and methods for the automatic generation of the symbolic equations and their numerical integration will be discussed.

Focusing again on the impact phase with high accelerations the concept of the wobbling mass will be introduced which allows for a physically more realistic simulation. The computational complexity involved can be motivated analyzing the structure of the equations of motion for the rather simple three-linked wobbling mass model. Based on a five-link model animation sequences will be shown performing various down jumps and summersaults.

4. Interactive control of direct dynamics

An approach still used quite often when trying to do a dynamic simulation of a complex motion is to derived the external forces and the internal torques from a real motion by means of film analysis or acceleration measurements. In contrast, direct dynamics prescribes the time development of the internal torques in the joints, which are generated by the skeletal muscles and thus reflect the free will of the being to control its motion. Although, while sophisticated active feedback mechanisms can be designed to control the decrease and the increase of the internal torques for example during a down jump, there remain many parameters which can only be determined by trial and error. This leads to the problem of an interactive control of a compute intensive simulation.

We use a remote procedure call mechanism developed for the DFN (German Research Network) to partition the problem into the numerical integration and matrix inversion which are performed by a compute server and the visualization of the motion performed on the graphics workstation. In order to allow for user interaction at the workstation even during the remote procedure call executing on the compute server a special shared memory server process is designed which is controled from the graphics application via named pipes and signals. Thus, the internal torques can be modified during the simulation with the attached dials and the compute server will use these updated values in its next time step.

5. Conclusion

The ultimate goal for the physically correct motion of multi-body systems is a task level control which derives the necessary internal torques from a data base which was filled by learning (e.g. under the control of a neural network) and by reacting to external feedback (e.g. vision). However, until then, a lot of trial and error processing will be necessary, demanding for an interactive control and realtime visualization of the dynamic simulation.

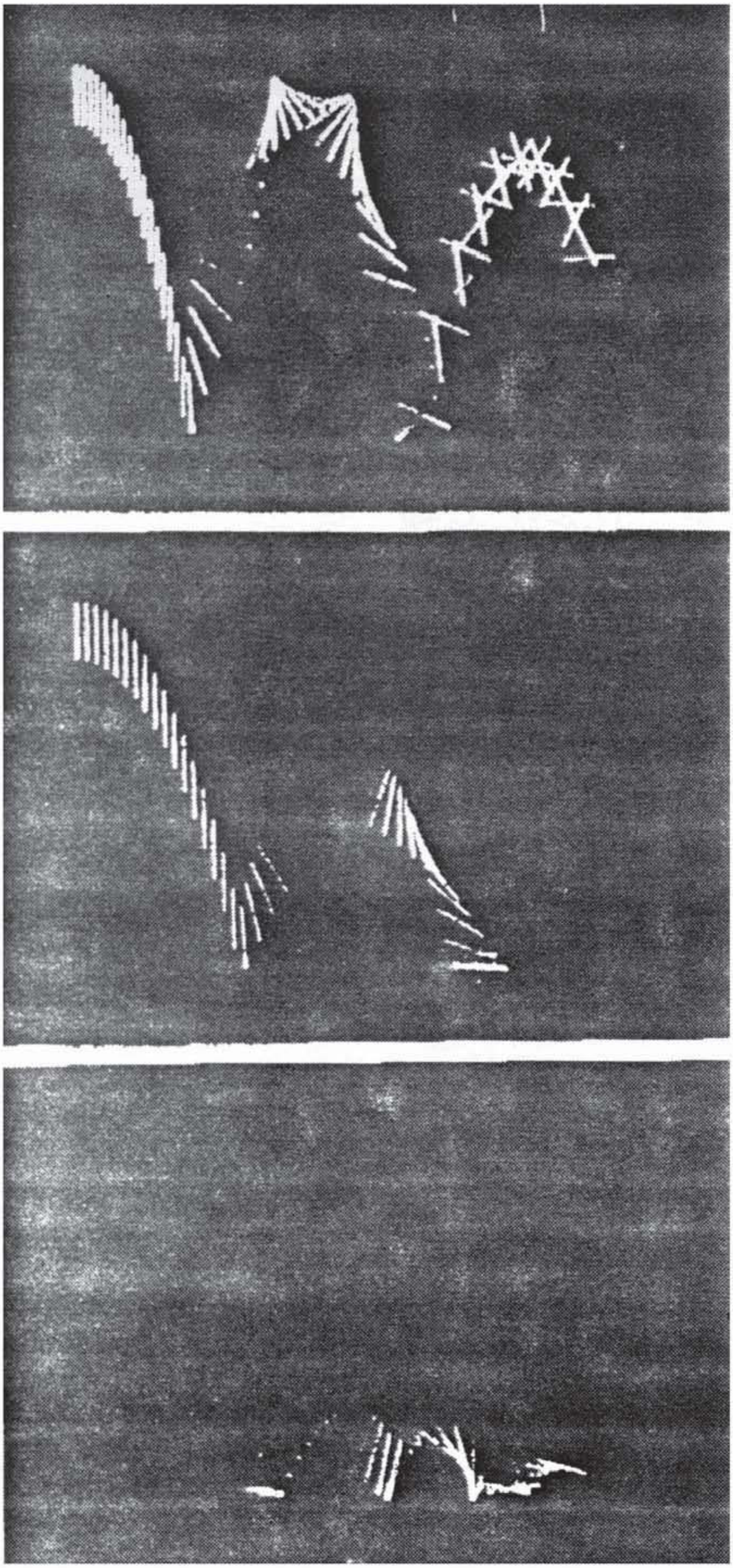

Fig 3.6 Three stroboscopic time series of a falling rod for an increasing (top to bottom) damping component in the ground reaction force.

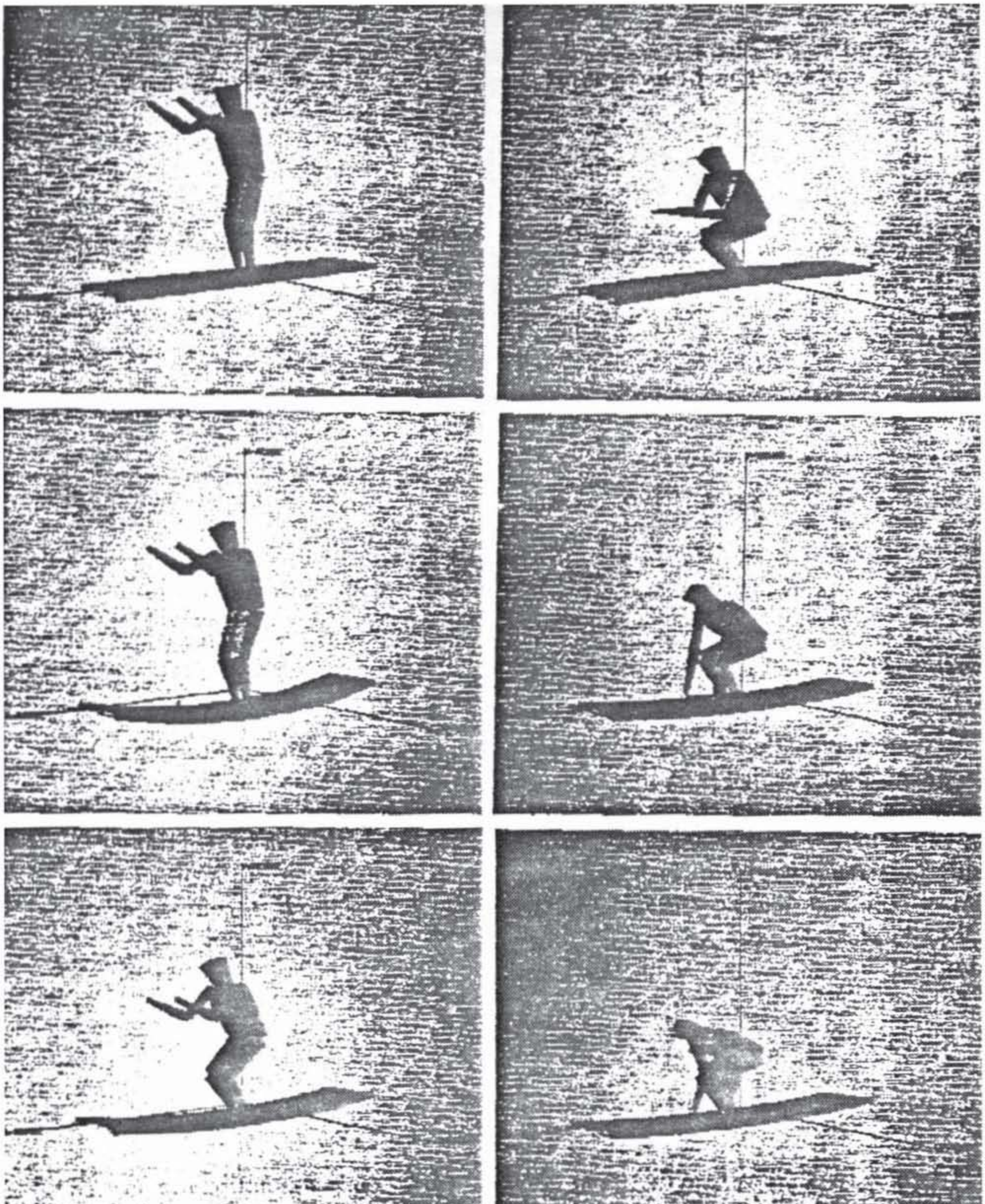

Fig. 5.8. Snapshots of a down jump of a five-linked model with landing on a bending plank. The internal torques are controlled in such a way that the model comes to rest in a squat position.

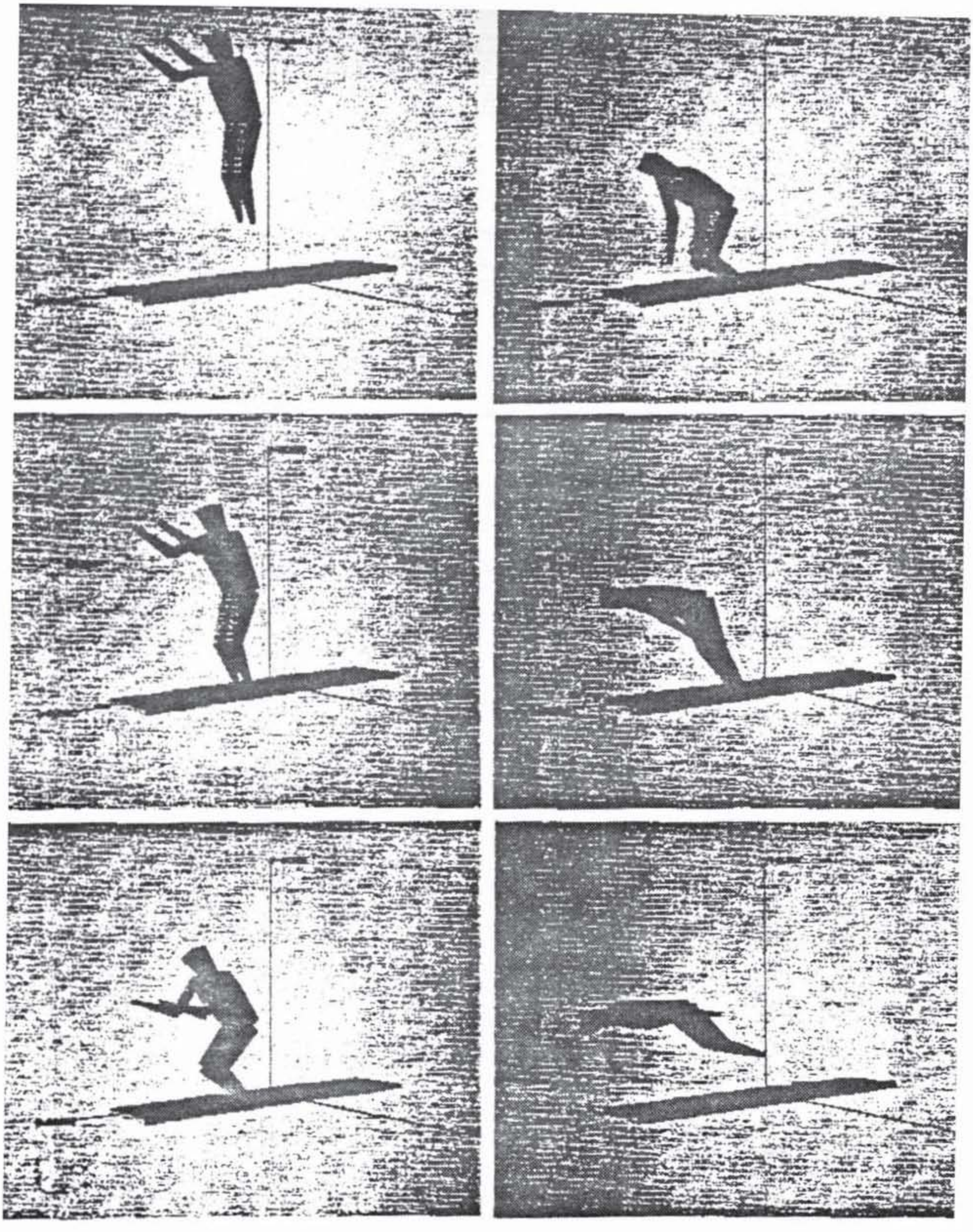

Fig. 5.9. Snapshots of a down jump of a five-linked model with landing on a stiff plank. The internal torques are here controlled in such a way that the model jumps off again and performs a somersault.

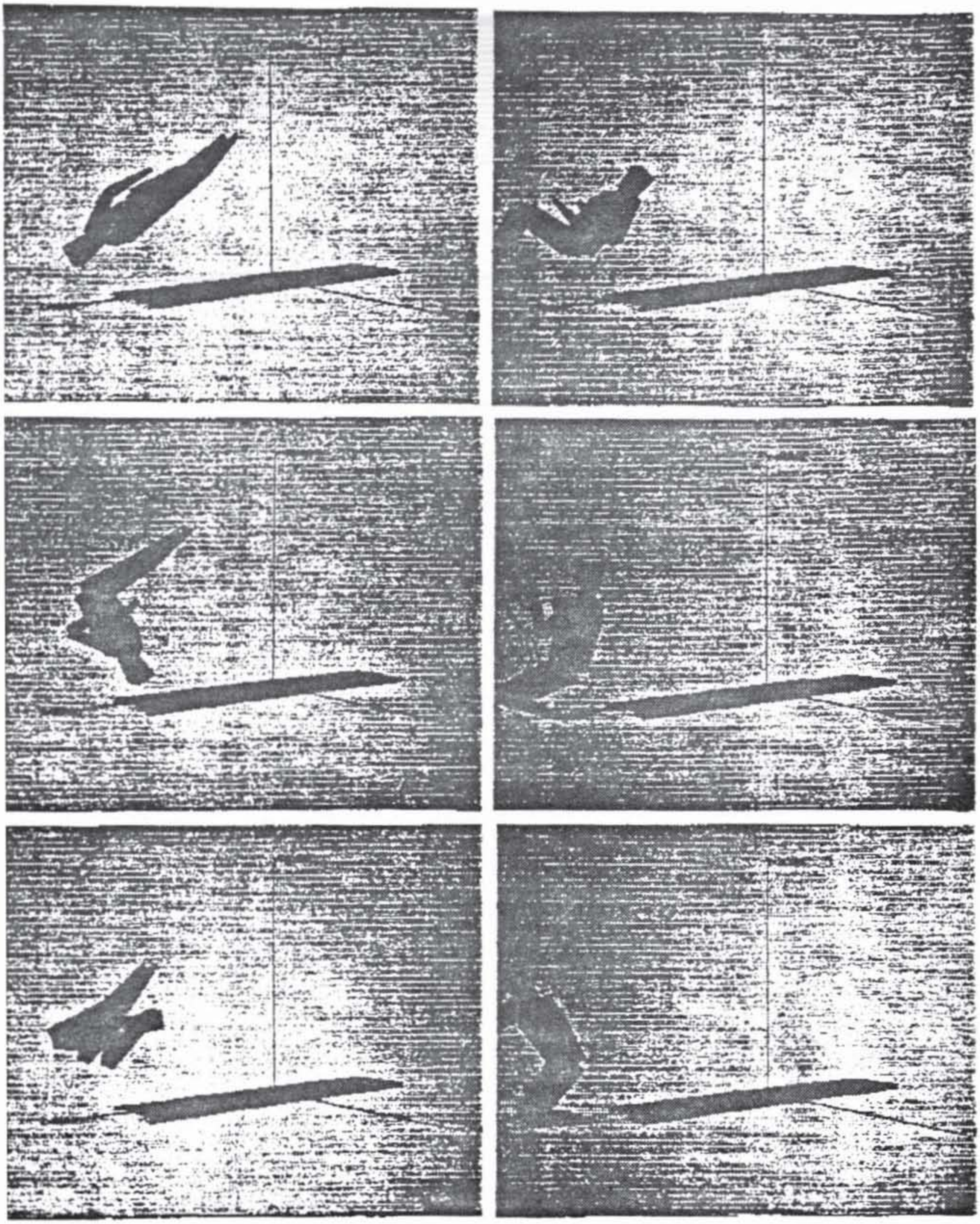

Fig. 5.9. (continued)