

FISH MOTION SIMULATION

Ievgen Smielik, Klaus Müller and Klaus-Dieter Kuhnert
Institute for Real-Time Learning Systems, University of Siegen
Hölderlinstr. 3, D-57076 Siegen
Germany
ievgen.smielik@uni-siegen.de

KEYWORDS

Estimation, tracking, practical implementation

ABSTRACT

In this paper we describe the retrieval of some real fish properties to establish them on a virtual fish. These include swimming motions. We introduce a special function which extracts a definite description of a fish. We use polynomial interpolation and spline to estimate fish body changes (moves), then reproduce such on a display.

INTRODUCTION

This paper describes work on a project called "Virtual fish". It is a cooperation between biology and computer science departments. The goal of the project is to achieve a virtual fish, that can visually interact with a real fish. We set a display showing a virtual fish next to aquarium with a real one, then alive fish should acknowledge an artificial one and both act accordingly. Fish visualization was implemented already. In short, we use a textured 3-D model with a sequence of N connected line segments (or bones), that represent fish backbone. Each bone can move, change its angle with another bone that it is connected with. All the bones can move in only one common plane. Each line segment has length, approximated from real fish images. Using our model allows us to draw a virtual fish on a screen to simulate a real fish. The central question of biology research here is a mate-choice, and as there are many aspects, we keep the system as flexible as possible: bone count, fish look can change.

Focus of this paper is on estimation of a fish swimming movement and its application to a virtual model. Described method is independent of fish species. We give details on how to describe and simulate a natural process on a computer.

To study fish behavior we set up a stand with aquarium and top view high speed camera. We record hours of video of fish behavior in aquarium for later analyze.

There are several works on fish simulation, authors of (Kim 2007) are mostly focused on a system performance in visualization, they also use a skeletal fish model.

There are authors using a robotic fish for simulations. Work (Ryuh 2013) is focused mainly on a robot tracking. Reference (Taverna 2010) has a main focus in a water flow around fish body and (Han 2008) has as well hydrodynamics as a basic topic.

Authors of (Terzopoulos 1994) use reinforced learning to create an artificial fish, which is an extensive task, our goal at presented stage is a motion simulation. We use a function approximation to reproduce actions of a real fish on a display.

Multi-joint approach is presented in (Qinyuan Ren 2014), authors combine oscillator with artificial neural networks to generate desired motion patterns.

IMPLEMENTATION

Fish model

We use our model with bones for both real fish tracking and virtual fish representation. On the one hand we map our model and a real fish, and on the other hand we display an achieved model.

Find fish area

To find a region where a fish is located we use segmentation technique described in (Klaus Müller 2014), basically background subtraction. Applied to an image with a fish, it results in a set of pixel coordinates where a fish is located, a fish area.

Find middle curve

With a fish area we have no definite location of a fish and no grip on finding bones, while we have to find the best possible location for our model bones. We transform a plain fish area to a set of points that describe a fish position exactly. As such points we selected a middle curve, we find locations inside a fish area evenly spaced between two fish sides. Fish sides are two parts of a fish contour, we find a contour with approach from (Suzuki and Abe 1985) and divide it in two pieces from head to tail and from tail to head. Location of head or tail we call a vertex on analogy with parabola vertex. First we find an approximation for vertex, to do that we fit the

whole fish area into bounding rectangle and find left-most and right-most points, these become both vertex estimates.

We compute a fish contour $C = \{C_k\}, k \in [1, P]$, which is a sequence of P points, where C_h and C_t are the nearest to estimated vertexes and $h < t$. h and t indexes split the fish contour in two parts: from head to tail and from tail to head, we define

A_{app} as $A_{app} \ni C_k, k \in (1, h) \cup (t + 1, P)$

and

B_{app} as $B_{app} \ni C_k, k \in (h + 1, t)$.

A fish area contour near a vertex represents a part of parabola, but fish sides become almost parallel near or behind side fins. We use that fact to find the precise vertex location. We note that for *some* points P from A_{app} the closest point in B_{app} is the first one. Points P lay on a parabola, but when two side curves go parallel then the closest point is not the first, let this point index be r . **Fig. 1** shows the approach graphically.

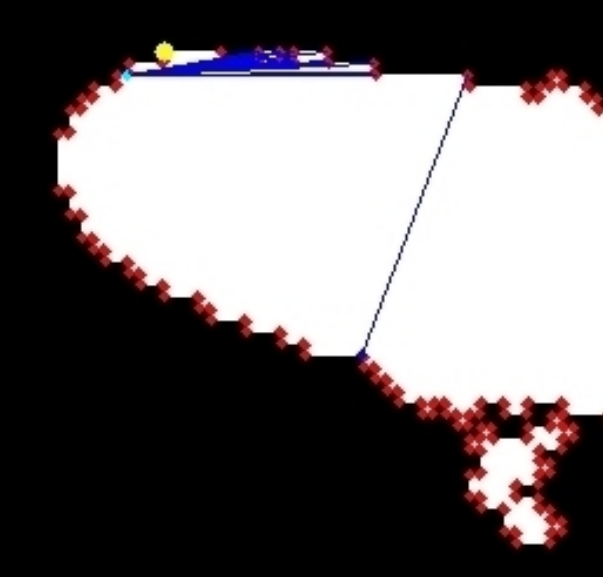


Figure 1: Vertex search. White is a fish area, black is background, red circles mark fish contour, blue lines connect minimum distant points on different sides, yellow circle is an approximation for vertex, cyan is a first point from A_{app}

We apply this method to find parabola end on other side and get point index l . Then real vertex index is the middle between l and r , $v = \frac{l+r}{2}$, point C_v . And real contours A and B achieved. We need to find the middle between point sets A and B , first we filter point sets, we have to remove fish side fins (which are also in a fish area). The issue here is that side fins are more or less transparent, but sometimes with a different angle or light conditions fins appear in a fish area, and we came up with a following approach, filtering takes a point from one side if it is at the minimum distance from some point

from another side,

$minA = \{A_k \in A\} : \exists B_m \in B, d(A_k, B_m) < d(A_i, B_m), A_i \in A$

and

$minB = \{B_k \in B\} : \exists A_m \in A, d(B_k, A_m) < d(B_i, A_m), B_i \in B$,

we achieve two smaller sets, having eliminated unwanted points, mostly fish side fins and also minor reflexions, and then we compute the fish middle curve, it contains middles between minimum distanced points from $minA$

and $minB$. Let $middleCurve = \{P_k\}, P_k = \frac{A_m + B_k}{2} : A_m \in minA, B_k \in minB, d(A_m, B_k) < d(A_m, B_i), B_i \in minB$. The result of finding a middle curve is shown on **Fig. 2**.

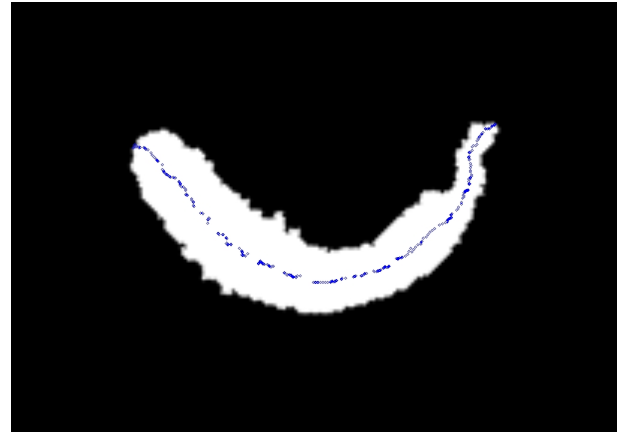


Figure 2: Fish middle curve

Estimate bones' angles

Till now we got fish middle curve $\{P_k(x, y)\}, k \in [1, C]$, we divide it into pieces, which lengths correspond to bones length. Let our bones sizes be $l_m, m \in [1, N]$. We choose such a set $S_m = \{P_k\}, k \in [r, p], m \in [1, N]$ of subsets of $\{P\}$, that sum of length between consequent points from one set, $\|S_m\| = \sum_{i=r}^p d(P_i, P_{i+1})$, equals to one bone size, $\|S_m\| = l_k$ and every bone has such a set of points. Then we fit a line into each bone's points. Line equation is $k * x + b = y$, by judging k and b as variables and taking x and y from point coordinates, we get an overdetermined system of linear equations that we solve in terms of least-squares problem with singular value decomposition method. As a result we have k and b line coefficients for each bone. Bones are shown on **Fig. 3**.

ESTIMATE BODY MOVING

Now we can track the fish and extract information about its motion.

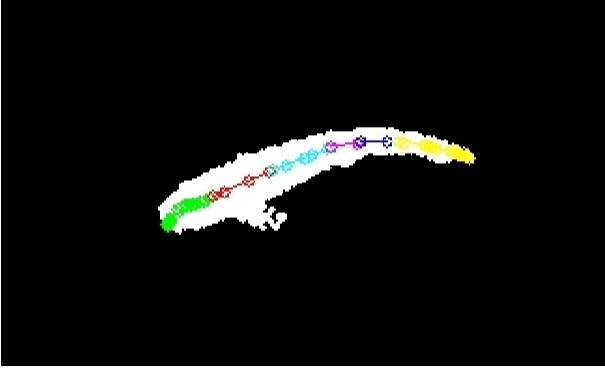


Figure 3: Bones. Each one is a line segment colored differently.

Find line trajectories

We search our fish videos for periods where fish moves mainly in one direction. We use our line fitting routine to check if a series of a consequent fish locations (taken from following frames) are nearly at one line. **Figure 4** shows one such period of linear swimming, full fish middle curves are plotted. **Figure 5** gives a different view.

There are three dimensions for our fish model: axis

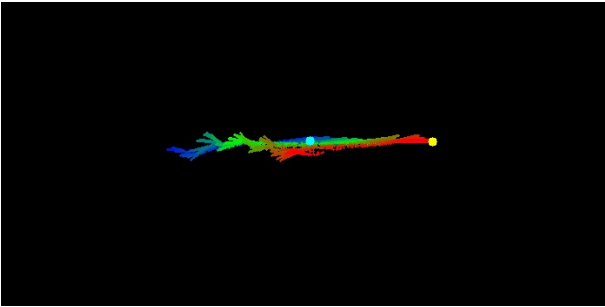


Figure 4: Line moving trajectory. Color change from blue to red shows frame number change from first frame to the last.

parallel body length, axis perpendicular to body length (from top view, describes mostly tail displacement) and time. On **Figure 4** the time is shown in different colors, two other dimensions are horizontal and vertical. Fish can bend in only one plane, turn a tail from side to side, so we need only two axis to describe fish body state. We note that any point on a fish tail follows sinusoid, so we use sinusoidal function in time dimension, while polynomial represents fish body bending. That leads to

$$y(x, t) = P(x) * \sin(B * t), \quad (1)$$

where $x \in [0, L]$ is body length measure, L is estimated 200, t is time (in terms of a frame number), $P(x)$ is a polynomial function, value of B affects the period, estimated value is 0.4, function result, y is a fish body displacement.

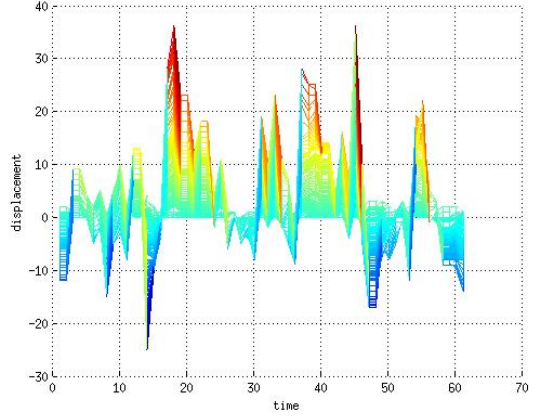


Figure 5: Line trajectory mesh. Body length axis goes against the viewer

Interpolate body curve

We use a general polynomial function equation in form

$$P(x) = \sum_{i=0}^p c_i * x^i, \quad (2)$$

where p is a polynomial degree. We want to find coefficients for such curve that fits our fish middle curve, we solve a system of linear equations. Number of equations is amount of points in middle curve, C , coefficients are unknown and we take x and y values from middle curve points coordinates. In that way

$$\begin{cases} c_0 + c_1 * x_1 + c_2 * x_1^2 + \dots + c_p * x_1^p = y_1 \\ c_0 + c_1 * x_2 + c_2 * x_2^2 + \dots + c_p * x_2^p = y_2 \\ \dots \\ c_0 + c_1 * x_C + c_2 * x_C^2 + \dots + c_p * x_C^p = y_C \end{cases} \quad (3)$$

We solve the system against unknown coefficients and get the best fit approximation to fish middle curve. But we note that not all the middle curves are good representable with a single polynomial, we divide fish movement in two periods in time: first period, when tail moves from zero to a point of the maximum amplitude and second period, when it moves back to zero. Fish starts to move from a zero point, when all bones lay nearly at one line, middle curves at different time at the period are concave. Plot of estimated polynomial is presented on **Figure 6**. After it reaches the maximum amplitude, middle curve functions become convex, one such plot is given on **Figure 7**. On the plots, horizontal axis represents fish body length while vertical shows bone displacement. Of course, there is only a finite number of bones, so we fit line segments into a curve and calculate angles between them.

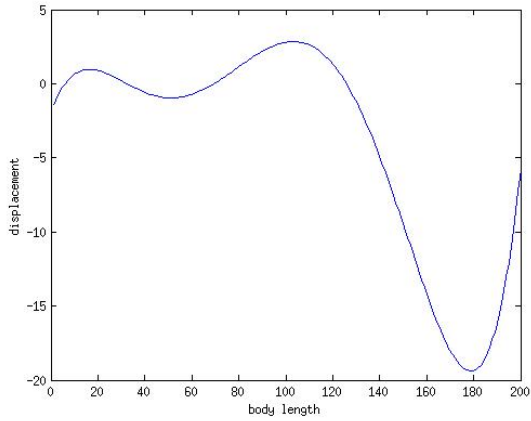


Figure 6: Concave polynomial

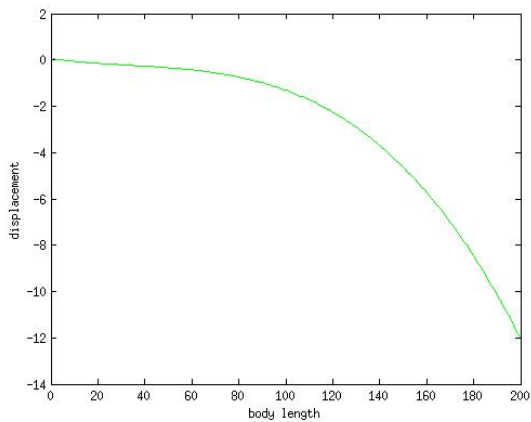


Figure 7: Convex polynomial

Interpolate between two polynomials

In order to change middle curve representation from one polynomial to another, we first used a third polynomial. Coefficients of a third polynomial are averages of corresponding coefficients of another two polynomials. **Figure 8** shows concave polynomial on top, convex at the bottom and an average one in the middle.

If we simply define time frames when to use which polynomial then moving function is non-continuous. **Figure 9** shows transition between polynomials with definite time frames. On this plot blue areas show values of a first period polynomial, green areas show values of a second period polynomial, red areas - an average one.

To improve fish movement we change from defining another polynomial to finding mean values. **Figure 10** shows transition between polynomials with average values.

We have used interpolation in fish length dimension to approximate middle curves, and now to achieve continuous motion function of body length measure and time, we use spline interpolation in time dimension, **Fig. 11**.

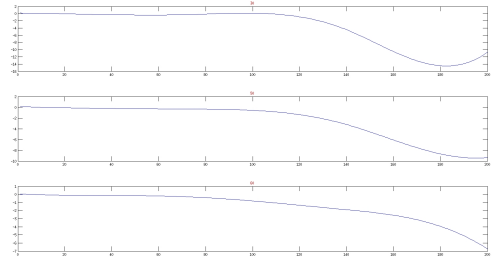


Figure 8: Transition between polynomials

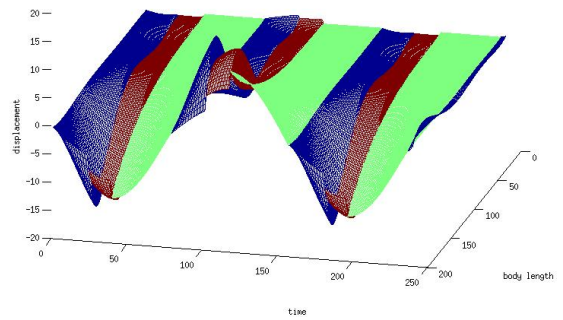


Figure 9: Step between polynomials

Here we cannot split the plot in polynomial representation parts as direction of spline curves is orthogonal to body length. There are as many splines as how long we choose the fish body to be.

CONCLUSION

In the presented work we show how to copy a real fish movements for use in computer environment. Behavior of achieved virtual fish is realistic as confirmed by biology expert. Our method base is to find a function that describes an object (or a process) and analyze the function. Major steps of current work include finding a middle curve, approximate it with polynomial function, build spline from values of polynomial. Middle curve of a fish is found using fish edges. Approximation of middle curve gives two polynomial functions, which we combine with a help of a cubic spline approximant. Approach is not restricted to neither fish behavior nor specific set up, and later research is needed to discover other choices of using proposed method of a surface estimation.

REFERENCES

Han D.X.W.C.J.L.L., 2008. *Simulation Study on the Body Side-Sway Characteristic for Rigid Robot Fish.*

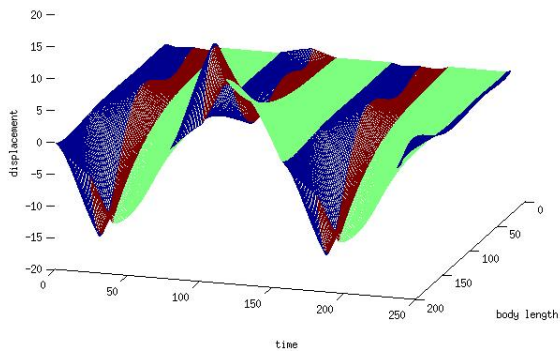


Figure 10: Gradual change between polynomials

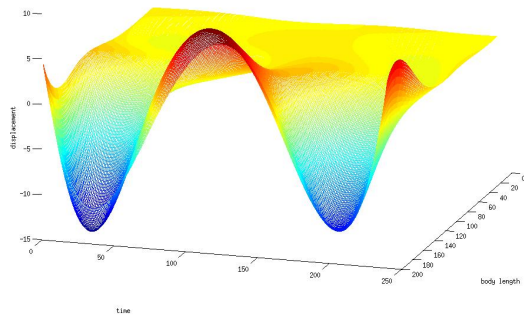


Figure 11: Spline two polynomials

Suzuki S. and Abe K., 1985. *Topological structural analysis of digitized binary images by border following*. In *Computer Vision, Graphics, and Image Processing*. IEEE.

Taverna Laura; Chellali R.R.L., 2010. *3D simulation of robotic fish interactions with physic-based underwater environment*. In *OCEANS 2010*. IEEE.

Terzopoulos Demetri; Tu X.G.R., 1994. *Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world*.

In *Robotics, Automation and Mechatronics*. IEEE.

Kim CheolMin; Shin M.J.S.K.I.K.E.K.B., 2007. *Real-Time Motion Simulation of Artificial Fish for Virtual Marine World*. In *Advanced Language Processing and Web Information Technology*. IEEE.

Klaus Müller Jens Schlemper L.K.K.D.K., 2014. *Calibration and 3D ground truth data generation with orthogonal camera-setup and refraction compensation for aquaria in real-time*. In *VISAPP 2014 - International Conference on Computer Vision, Theory and Applications*. IEEE.

Qinyuan Ren Jianxin Xu Z.G.Y.R., 2014. *Motion Control of a multi-joint robotic fish based on biomimetic learning*. In *Industrial Electronics (ISIE), 2014 IEEE 23rd International Symposium on*. IEEE.

Ryuh S.C.L.Y.T.K.S.K.H.Y.S., 2013. *Simulation based design for position estimation of small robotic fish*. In *Ubiquitous Robots and Ambient Intelligence (URAI)*. IEEE.