~~3/1/18~~

3/1/19

~~Interpolating:-~~

3/1/19

Interpolation:- of position of point in space

- Appropriate Interpolating function.
- parameterization of function based on distance travelled.
- Desired Control of Interpolated position over time.

(-5,0,0) (5,0,0)

22 67

stop stop

(-5,0,0) (5,0,0)

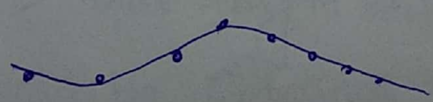
22 → 34 ← 67

accelaration max speed stop

67 → 80.

Commonly Used Interpolating functions:-

- Hermite formulation — tangent information at end point.
- Catmull Roam Spline — positions curve should pass through



~~9/1/18~~
4/1/19

Controling motion of a curve ~~is traced~~

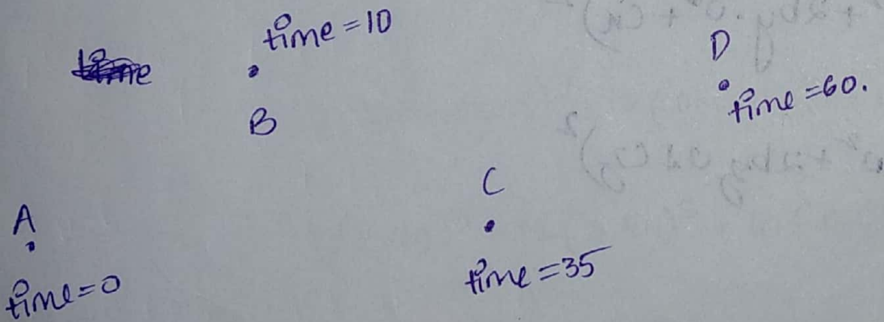
* Designing Shape of a curve
* Speed at which curve is traced — parametric value
  ↓
  direct Control
  ↓
  predict result.

9/1/19

Time-position-~~pairs~~ pairs:—

(0, A) (10, B) (35, C) (60, D)

↓ time ↳ position



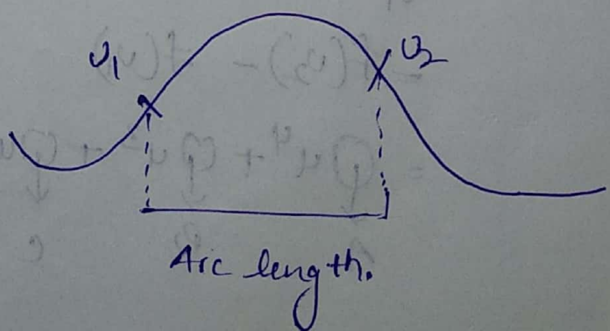
$P(u) = ((x(u), y(u), z(u)))$

$p(u) = au^3 + bu^2 + cu + d.$

$x(u) = a_x u^3 + b_x u^2 + c_x u + d_x.$

$y(u) = a_y u^3 + b_y u^2 + c_y u + d_y$

$z(u) = a_z u^3 + b_z u^2 + c_z u + d_z.$



Analytic approach to Compute Arc length:-

$$S = \int_{u_1}^{u_2} \left|\frac{dp}{du}\right| du \quad , \quad \frac{dp}{du} = \left(\frac{dx(u)}{du}, \frac{dy(u)}{du}, \frac{dz(u)}{du}\right)$$

$$\left|\frac{dp}{du}\right| = \sqrt{\left(\frac{dx(u)}{du}\right)^2 + \left(\frac{dy(u)}{du}\right)^2 + \left(\frac{dz(u)}{du}\right)^2}.$$

$\Rightarrow$

$$x(u) = a_x u^3 + b_x u^2 + c_x u + d$$

$$\frac{dx(u)}{du} = 3 \cdot a_x \cdot u^2 + 2 \cdot b_x\, u + c_x.$$

$$\left(\frac{dx(u)}{du}\right)^2 = \left(3a_x \cdot u^2 + 2b_x\, u + c_x\right)^2$$

IIIly $$\left(\frac{dy(u)}{du}\right)^2 = \left(3a_y \cdot u^2 + 2b_y \cdot u + c_x\right)^2$$

IIIy $$\left(\frac{dz(u)}{du}\right)^2 = \left(3a_z u^2 + 2b_z u + c_z\right)^2$$

$$\int_{u_1}^{u_2} f(u)\,du$$

$$= f(u_2) - f(u_1)$$

$$= \underset{A}{\underset{\downarrow}{Q}} u^4 + \underset{B}{\underset{\downarrow}{Q}} u^3 + \underset{C}{\underset{\downarrow}{Q}} u^2 + \underset{D}{\underset{\downarrow}{Q}} u + E$$

$A = 9(a_x^2 + b_y^2)$ $\quad B = 12(a_x \cdot b_x + a_y \cdot b_y)$ $\quad C = 16(a_x c_x + a_y \cdot c_y) + 4(b_x^2 + b_y^2$

Scanned by CamScanner

$D = 4(b_x \cdot c_x + b_y \cdot c_y)$
$E = c_x^2 + c_y^2.$

10/01/19

Foward differencing:- (Computing Arc length)

$G(0) = 0.00 \qquad G(u) = p(u)$

$0 - 0.5$

$G(0.5) - G(0)$

$P(0) - p(0.5)$

$$\int_{u_1}^{u_2} f(u)\,du = \int_{0.0}^{0.05}$$

$= 9(0.0-0.05)^4 + 12(0.0-0.05)^3 + 16(0.0-0.05)^2 + 4(0.0-0.05) + 1$

$= 9(0.95)^4 + 12(0.94)^3 + 16(0.95)^2 + 4(0.95) + 1$

$=$

$=$

1

Scanned by CamScanner

Index	Parametric Entry	Arclength
0	0.00	0.000
1	0.05	0.080
2	0.10	0.150
3	0.15	0.230
4	0.20	0.320
5	0.25	0.400
6	0.30	0.500
7	0.35	0.600
8	0.40	0.720
9	0.45	0.800
10	0.50	0.860
11	0.55	0.900
12	0.60	0.920
13	0.65	0.932.

$$i = (\text{int})\left(\frac{\text{given parmetric value}}{\text{distance between entires}} + 0.5\right)$$

given parametric value $= \left(\frac{0.42}{0.05} + 0.5\right)$

$= 8.9$

$= \text{int}(8.9)$

$= 9.$

Can be approximated $i = (\text{int})\left(\frac{\text{given parametric value}}{\text{distance between entries}}\right).$

Scanned by CamScanner

25/01/19

$$L = Arclength[i] + \left| \frac{(givenValue - value[i])}{(value[i+1] - value[i])} \cdot (Arclength(i+1) - Arclength(i)) \right|$$

to= Arclength

$$l = 0.720 + \left| \frac{0.42 - 0.40}{0.45 - 0.40} \cdot (0.800 - 0.720) \right|$$

$$= 0.720 + \frac{0.02}{0.05}(0.080)$$

$$= 0.720 + 0.032$$

$$= 0.752.$$

31/01/19

Speed Control:-

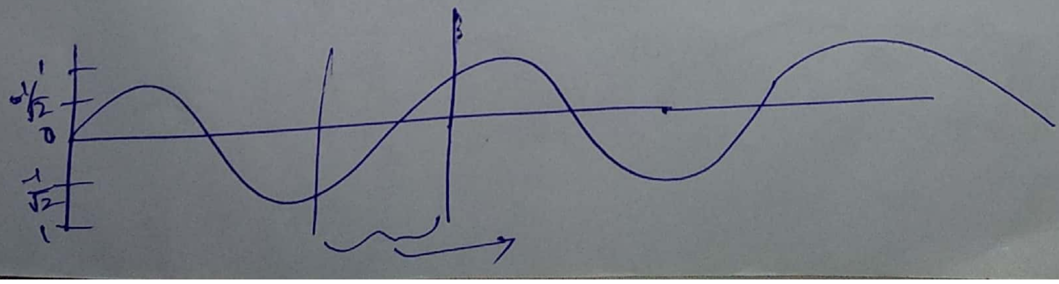
distance time

$s(t) \rightarrow$ distance w.r.t time.

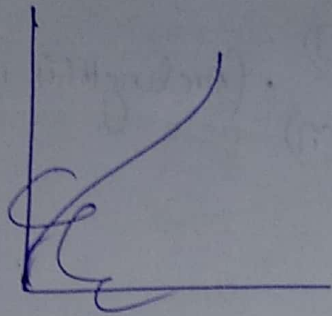
Constant Speed:- distance - equal - intervals

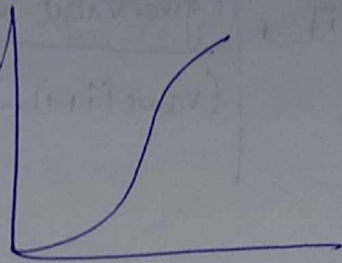
Ease-in / Ease-out:-

$$s(t) = ease(t) = \frac{\sin\left(t \cdot \pi - \frac{\pi}{2}\right) + 1}{2}$$



Sine wave.

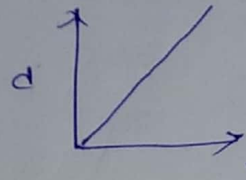




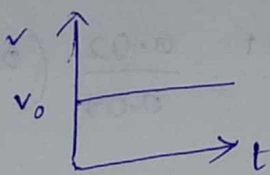
car(t)

01/02/17

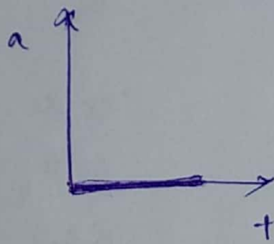
distance $d \propto t$



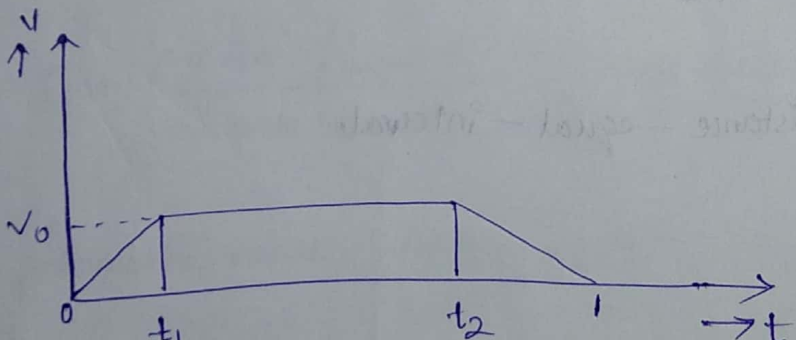
velocity



Accelaration :-



Velocity :-



(a) distance Covered by area under the Curve

distance ~~Covered the~~ object $= \frac{1}{2} t_1 v_0 + (t_2 - t_1) v_0 + \frac{1}{2}(1 - t_2) v_0$.

distance – time function

$$d = v_0 \cdot \frac{t^2}{2t_1} \qquad 0 < t < t_1$$

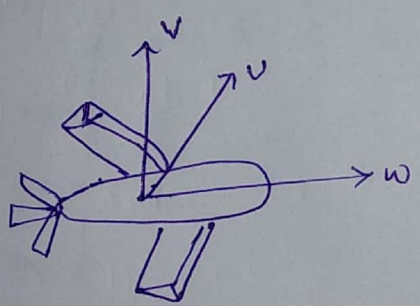
$$d = v_0 \cdot \frac{t_1}{2} + v_0(t - t_1) \qquad t_1 < t < t_2$$

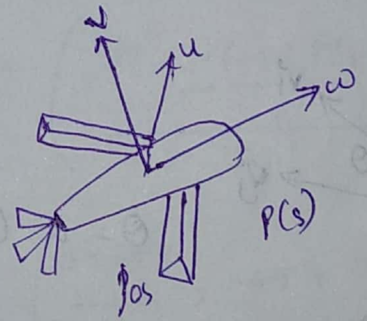
$$d = \left(v_0 \cdot \frac{t_1}{2} + v_0(t_2 - t_1) + v_0 - \left(v_0 \cdot \frac{t - t_2}{1 - t_2}\right)\right)(t - t_2) \qquad t_2 < t < 1.0$$

05/02/19

path following:–

frenet Frame:–





w – view vector

v – up vector

u is ⊥ar to both 'v' and 'w'

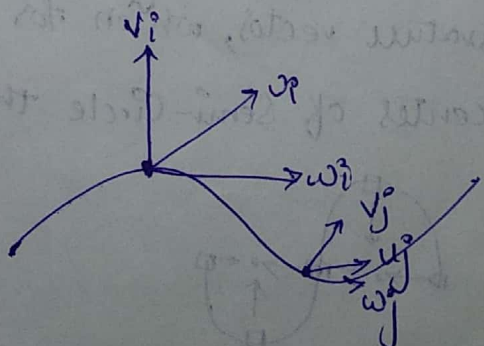
$$\frac{d}{dx}\left(\frac{df(u)}{du}\right)$$

$p''(s)$ $\quad$ $p'(s)$

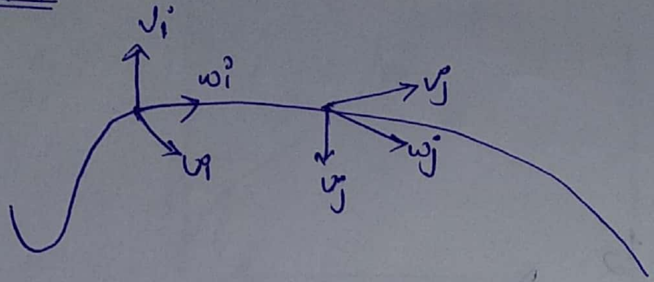
$$w = p'(s)$$

$$u = p'(s) \times p''(s)$$

$$v = w \times u$$

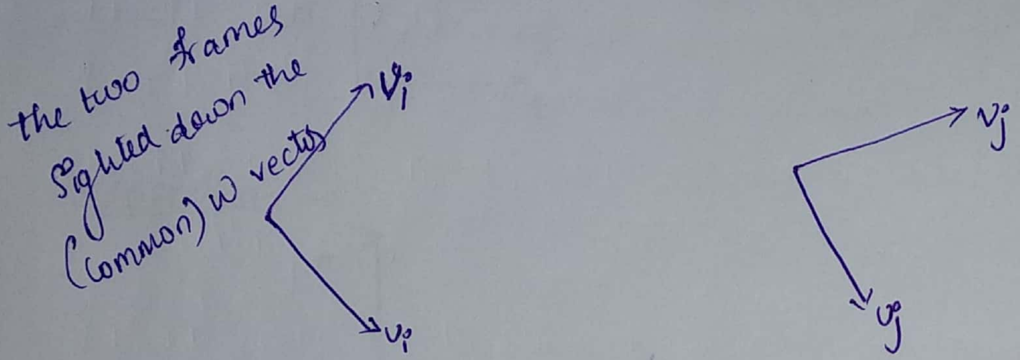


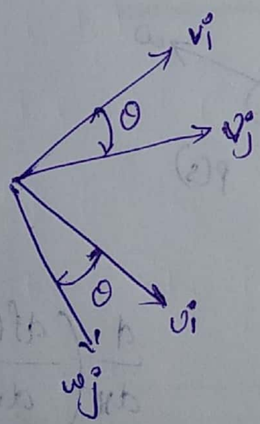
08/02/19



Frenet Frames on

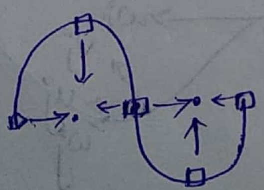
if the curve on which object is moving have no curvature, find orientation at boundaries.





$\theta = a\cos(v_1, v_2)$

two semicircles spliced together so, that they form an S(Sigmoidal) shape the curvature vector, which for any point along this curve will point to the center of semi-circle that point is on.



12/02/19

Camera path following:-

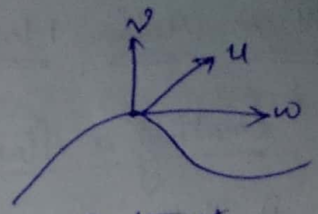
~~w = Cos-Pos~~ $w = COI - POS$

$U =$ wry-axis

$V = \bar{U} \times w$

COI - Centre of Interest

POS - position



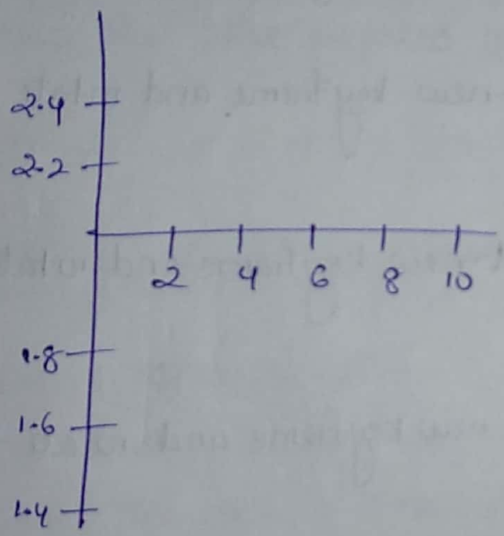
position of the object is $p(s)$ make $p(s + \partial s)$ as centre of Interest.

Consider the value of δs as low as possible.

Smoothing a path:-

(1, 1.6) (2, 1.65) (3, 1.6) (4, 1.8) (5, 2.1) (6, 2.2) (7, 2.0) (8, 1.5) (9, 1.3) (10, 1.4)

Smoothing with linear Interpollation of adjacent values.



let $p_i = (3, 1.6)$

$P_{i-1} = (2, 1.65)$

$P_{i+1} = (4, 1.8)$

$$P_i' = \left(\frac{P_i + \frac{P_{i-1} + P_{i+1}}{2}}{2} \right)$$

$$P_1' = \left(\frac{3 + \frac{2+4}{2}}{2}, \frac{1.65 + \frac{1.6 + 1.8}{2}}{2} \right)$$

$= (3, 1.66)$

Smoothing with cubic interpolation of adjacent values:-

19/02/19

Macro Media Flash player 8.0

Creating a flash movie-

1) Make story board ready
2) Open Macro media flash player, click on menu item - file - new - flash movie.
3) Create a new Symbol (draw a circle, divide it into 4, fill every part with a color).
4) Open timeline, right-click at zero and insert - new - key Frame.
5) Right click on timeline at 10 and insert - new - key Frame, and rotate Symbol by ~~45~~ 45°. (use free transform).
6) Right click on timeline at 20 and insert-new-keyframe and rotate the Symbol by 45°.
7) Right click on timeline at 30 and insert-new-keyframe and rotate the Symbol by 45°.
8) Right click on timeline at 40 and insert new keyframe and rotate the Symbol by 45°.
9) Right click on timeline anywhere between '0' and '45', select "create motion tween"
10) Click on the menu item - controls, select test movie
11) Save project (.fla) and export to ~~designed format~~. desired format(.swf, .gif, .avi - - -)

21/02/19

Creating a presentation in Macro Media Flash Player 8.0 :-

1) Click on the Menu item "File."
2) Select "New"

Scanned by CamScanner

3) Select presentation.

4) Import All media Files Required, into library. (File - import - import to library)

5) Open timeline place desired text/image at desired position. (~~insert new key frame at 0~~)

6) In the First slide, place the text "Bapatla Engineering College" at top middle position.

7) Click on Timeline at 5 and insert new key Frame and drag an image in library to the middle of the slide.

8) Right click anywhere between 0 and 5 on timeline and Select Create "motion tween"

9) Now the slide prepared can be tested. (Controls - play video)

26/02/19

making desired image as background:-

1) Open photoshop editor
2) ~~Create~~ drag image1 into photoshop
3) Select the desired area in a using move tool
4) drag the selected area of image2 on the image1.
5) place this at appropriate position.
6) using magic white tool ( ) crop, image2.

28/02/19

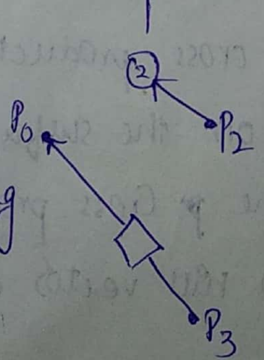
~~25/03/19~~

Smoothing with Convolution ~~Filtres~~ kernals:-

Convolution kernal

$$\boxed{P(x) = \int_{-s}^{s} f(x+u) \cdot g(u)\,du}$$

kernal Corresponding function



Scanned by CamScanner

05/03/19

1) polygonal Surface Mesh

2) greedy-type Algorithm

1) Construct a path of edges along a mesh surface from a given start vertex to a given destination vertex

2) For each edge emenating from current vertex (initially start vertex)

3) Calculate projection of each edge onto straight line between current vertex and destination vertex.

4) divide this distance by length of the edge to get the cosine of angle between edge and straight line.

5) the edge with largest cosine is the edge most in the direction of straight line.

6) choose this edge to add to the path.

7) keep applying this until the destination edge is reached

3) down Hill:-

if a path downhill from initial point on the surface is desired then,

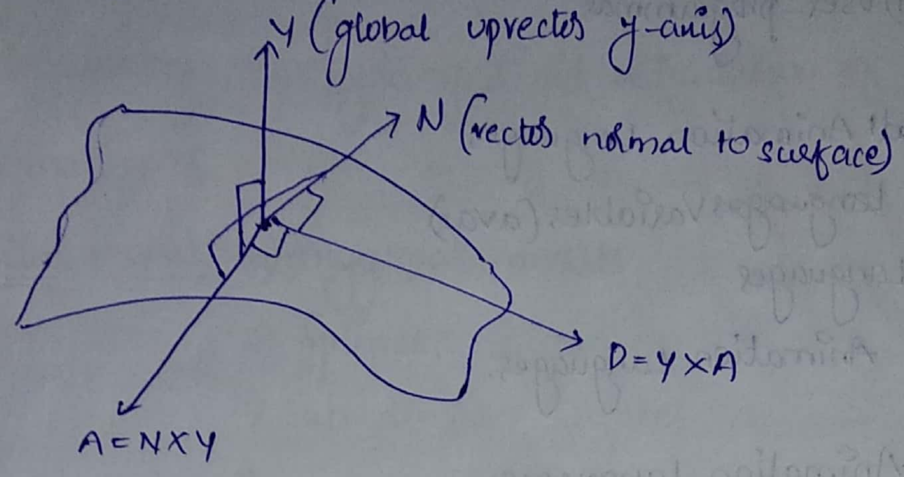
*) the surface normal and global up vector can be used to determine the down hill vector

*) the cross product of normal and global upvector defines a vector that lies on the surface perpendicular to down hill direction.

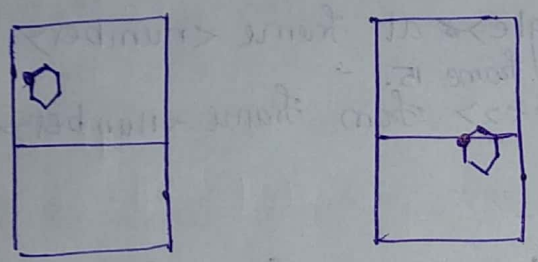
*) So the ~~p~~ Cross product of this vector and normal vector defines the down hill vector on a plane.

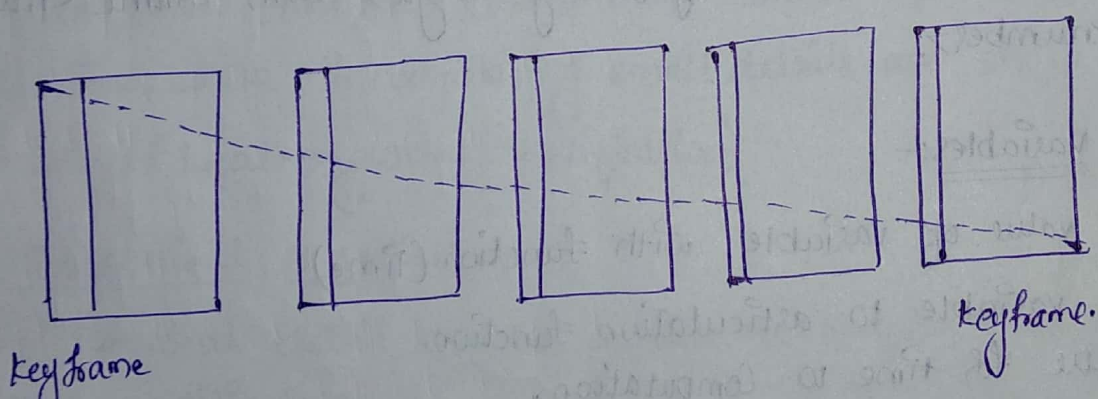
Scanned by CamScanner

dawn killing



Key Frame Systems:-





07/03/19

Animation Languages:-

⟹ First animation systems - ~~FOR~~ FORTRAN

* program ~~written~~ written in animation language - Script.

* Advantages:- 1) language hard coded - used any time - Regenerate it

2) powerful- interpolate values, Complex numerical qualities, behavioural rules

disadvantages:- 1) user-programmar

- ⇒ Artist Oriented Animation Languages
- ⇒ Articulation ~~Languages~~ Variables (avar)
- ⇒ Graphical Languages
- ⇒ Actor-Based Animation Languages.

⇒ **Artist Oriented Animation Languages:-**

⇒ set position <name><x><y><z> at frame <number>

⇒ Set rotation <name> [x, y, z] to<angle> at frame <number>
(set rotation <object> [10, 20, 30] to 45° at frame 15.

⇒ change position <name> to <x><y><z> from frame <number> to frame <number>.

⇒ change rotation<name> [x, y, z] by <angle> from frame <number> to frame <number>.

⇒ **Articulation Variables:-**

- ⇒ associating value of variable with function (time)
- ⇒ pass time variable to articulation function.
- ⇒ returns value for time to Computation.
- ⇒ this, technique goes by a variety of names, such as track channel (or) articulation variable

08/03/19

FACIAL ANIMATION

A good facial model must be capable of geometrically representing a specific person (Confirmation)

Scanned by CamScanner

used- realistic character animation, human computer interaction, tele communications.

Address- ~~lip-syncing~~ lip-syncing, rigid articulation of jaw and muscle deformation of tongue ----

types of facial models:- 1) polygonal models
2) Splines.
3) Sub-division Surfaces

1) polygonal Models:- smoothness of surface $\propto$ Complexity of Model.

⇒ relatively easy to create

2) Spline Models:- bezier (or) B-spline - represent face

⇒ Rectangular grid of Control points used, to model entire object.

⇒ Entire row and/or Columns of Control information must be modified, while incorporating small details and sharp localized features (local → global Modification)

⇒ Hierarchical B-splines

* local detail can be added to B-spline Surface without global modification.

3) Sub-Division Surfaces:-

⇒ Use polygonal control mesh that is refined, in the limit, to a smooth Surface, the refinement can be terminated at an intermediate resolution and rendered as a polygonal mesh.

Scanned by CamScanner

Creating the Model:-

⇒ Creating model of human head from ~~scrat~~ Scratch is not easy

⇒ geometric elements (vertices, edges) must be placed appropriately to control the motion of the surface.

⇒ if model is dense - low resolution, ~~use~~ cad ~~system~~ ask uses to construct model by CAD system

⇒ Besides CAD approach two main methods — Digitization using some physical reference (particular person desired)

— Modification of existing models. (animation control is already built in general model)

①

⇒ physical Sculpture of Desired object can be generated with clay, wood, (or) plaster and then digitized using mechanical (or) magnetic digitizing device.

⇒ A 2D Surface based Coordinate grid can be drawn on physical model and polygons can be digitized on a polygon-by-polygon basis

⇒ post processing can identify unique vertices, and a polygon mesh can be ~~easy~~ easily generated.

12/03/19

(2)
⇒ laser Scanners use a laser to calculate distance to a model surface and can create very accurate models
⇒ directly digitizes a person's face
⇒ Capture Colour information - generate texture map.
⇒ drawback - expensive, bulky and require a physical model

(3) ⇒ models can also be generated from photographs
⇒ not required presence of physical model, once the photograph has been taken
⇒ generating a model from stratch is to take front and side images of the face, on which grid lines have been drawn (Symmetry is usually assumed for the face, so only one side view is needed and only half of the front view is considered.

(4)
⇒ Modifying an existing model is a popular technique for generating a face model (Someone had to first generate a generic model)

Scanned by CamScanner

⇒ if it is created as a parameterized model, parameters were well designed, the model can be used to try to match a particular face.

⇒ Animation Controls can be built into model so that they require little (or) no modification of generic model for particular instances

<u>parameterized model Created by path</u>

⇒ parameters
- Confirmational (distinguish one individuals head and face from others)
- Expressive (animation of individual's face)

⇒ 5 parameters control shape

1) Forehead
2) Cheek Bone
3) Cheek Hallow
4) Chin
5) neck

⇒ there are 13 scale distances - facial features

1) head - x, y, z
2) chin to mouth
3) chin to eye
4) eye to forehead
5) eye x and y
6) widths of jaw, cheeks, nose bridge, nose nostril

⇒ 5 parameters translate features of face

1) chin in x and z.
2) end of nose x and z
3) eye brow z

Scanned by CamScanner