11.0

#2

Notes Starting 3/29/77

ending 6/3/77

VERNON
ROYAL

Sandy Ressler
RPO 7065
Rutgers College.

VERNON
ROYAL

COMPOSITION BOOK

60 SHEETS · 10¼ IN. x 7⅞ IN.

AVAILABLE AS:

No. 1148	(09-9142)	11/32″ WIDE RULED - NO MARGIN
No. 1148-CM	(09-9144)	9/32″ COLLEGE RULED - 1¼″ MARGIN
No. 1148-¼	(09-9148)	QUAD. RULED 5 SQS. TO INCH
No. 1148-½	(09-9146)	PLAIN
No. 1148-¾	(09-9140)	11/32″ WIDE RULED - 1¼″ MARGIN

Made in U. S. A.

VERNON ROYAL, INC. Elizabeth, N. J. 07208

Computer

Print out characters on Typeball Terminal
at random and document as much as possible

Time
Position
other B.S.

Use different color Typing ribbons (Aaron Marcus did this)

A Run should look like

Possible characters are: ....☆∂↗ ∿∿∿ = — ~

Random link is: xxxxxxx

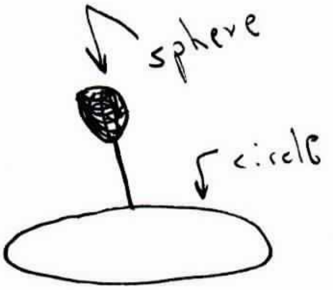
Δ # ←/¢ x y + ~ → ⊙

character	Time of selection for printing
Δ	1977 3 30 xxx xxx xx xxx
#	
←	
/	
¢	

---

character	spce Position in Line
Δ	5
#	9
←	12
/	16
¢	40

C. Graphics



Transformed into itself Rot x°, its Axis

But what's the point of it!
Don't do Just cute graphics.
(unless Theres money in it!!)

Need clipping for Apl
look and see how omnigraph does it

~~Additi~~ Dance

Additional features need for MAN

need rotation around body lines

I.e. rot Lower!Lleg x° around Thigh

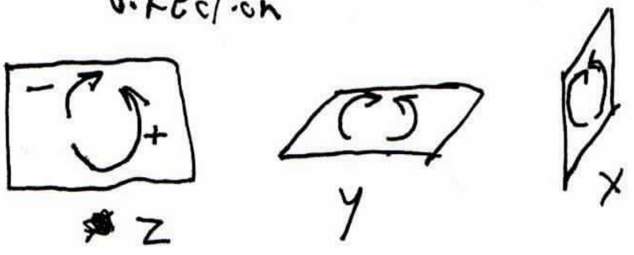
Thigh is a body line.

Reset for individual bod parts

sTatus Command

1. current veinpoint
2. " GeT RoT

need reference ~~circles~~ grids arrows to indicate + and - angle direction



Automation of Scus

```
∇ AutoScu

  count ← 1
  sculp ← Non Drawing sculp
  Fplot obJ

  Fend
  count ← count + 1
```

4/10

If ingredients are a <u>description</u> of a substance or they are the substance itself and a recipe is a specific ordering of these ingredients Then a program is the recipe which ~~[illegible]~~ orders the ingredients ~~(statement)~~ (input) and makes ~~[illegible]~~ a substance (output)

what is the relation of ingredients to the substance?

when combined they lose their individuality and become one (the substance)

a point loses its individuality when it is part of a whole drawing

The basic ingredient I deal with is the point ~~However~~ However ~~the~~ kinds of combinations of points lines, planes, cubes... becomes infinite.

IT is These combinations of ingredient, which are The real dealable ingred.

where does description fit in?

A program describes how To put together ~~a~~ substance

The Program is The description of The substance in relation To The input.

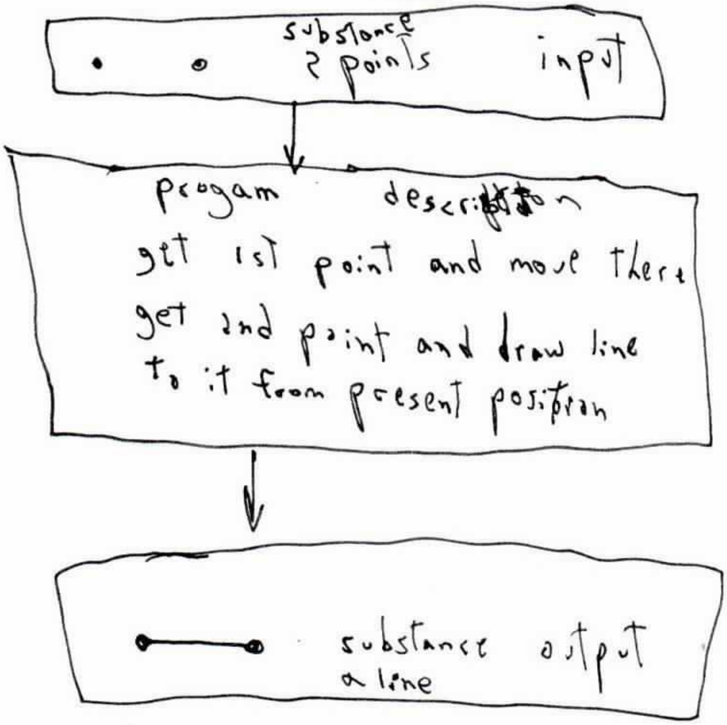
So in order To make The ~~program~~ desription ~~any~~ sensable to a reader The program must be put into words or some other method of describing it. Visual?

How do you describe a description?

A program itself becomes the input for another program .....ad infinitum

need some kind of program which becomes a subpart of another program which becomes a subpart of another.....

Modular programs which fit together and grow


substance
2 points
input
progam description
get 1st point and move there
get 2nd point and draw line
to it from present position
substance
a line
output

output is described By program but only in terms of input

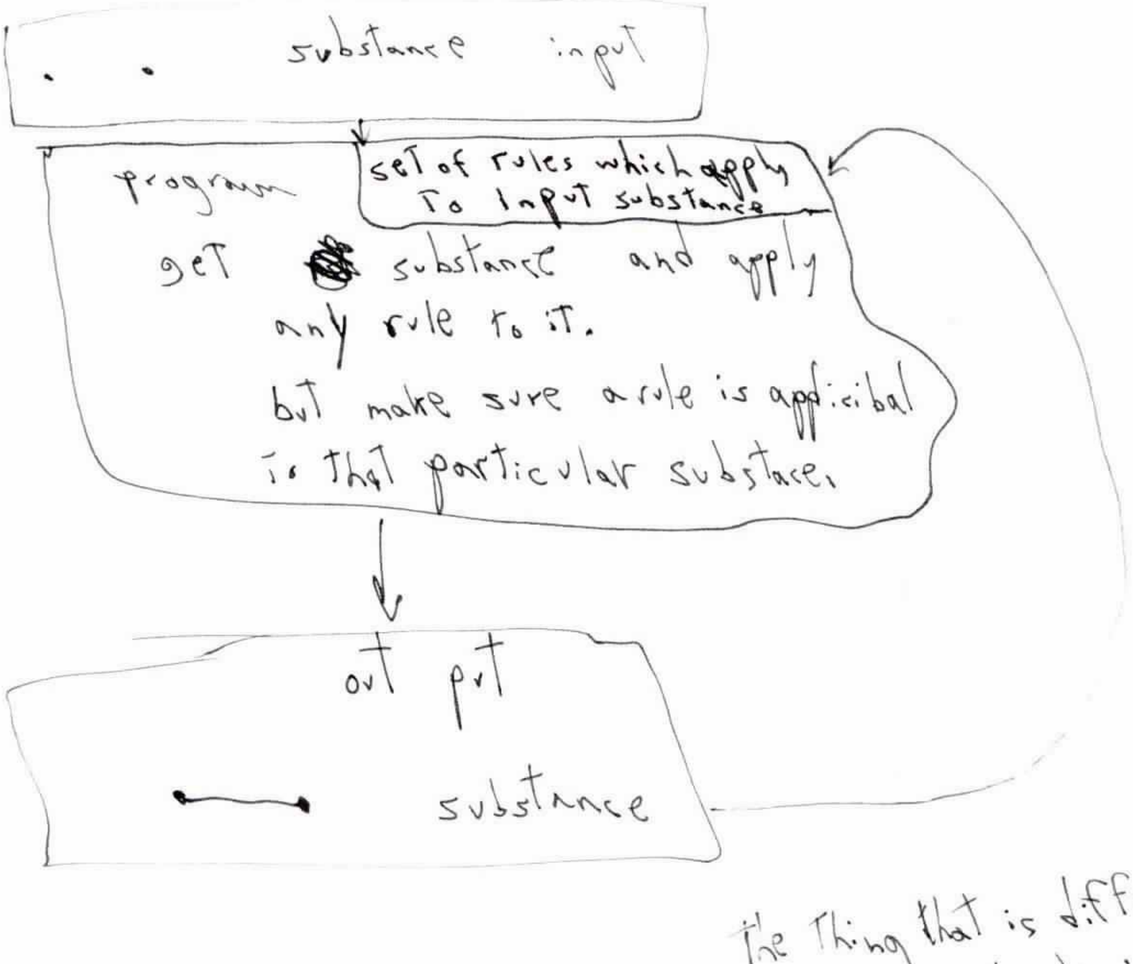
both input and output are substances so how can both be input into same program.

don't want same program but one which grows

you are getting

substances so it is a variable dependent upon the output of program.

a move or a draw in The program are arbitrary rules which happen To apply to points. but as The substace gets more complex more rules are possible and must be taken into account.



substance input

program

set of rules which apply To Input substance

get substance and apply any rule to it.

but make sure a rule is applicibal to that particular substace.

out put

substance

The Thing that is different each time you loop back is The set of rules which apply to a now different substance.

What are the rules and how many are there
and is there any limit?

Can the computer generated its own rules?

Rules for object generation

1. That apply to a point
   - A Move
   - B Duplicate

2 That apply To a line
   - A. Duplicate
   - B. Move
   - C. Scale
   - D. rotate
   - E. Split or Break
   - F. Bend
   - ~~G. Color~~

③ That apply To a surface
all of above plus
   - A Shade
   - B Texture
   - ~~C Color~~

④ That apply To a volume

~~Ⓐ [illegible] things [illegible]~~

~~Ⓑ~~

Maybe theres a very finit no of rules
(small)
and it is the combinations of rules which
are infinite.

And The Computer <u>can</u> generate Those combinations.

~~Need to Fi~~

Above rules are rules for something very specific
Computer graphics. What kind of other rules are There in
The world, life

① Observation (logical and emotional)

② Actions & reactions

Ⓐ Emotions

Ⓑ logic

The artist is
the emotional and
The computer can simulate
The logical.

Deal with rules that will produce some artifact

If gonna do computer art must deal with rules acceptable to computer, ~~and~~

An object can be described always in terms of a points by following some sequence of rules.

Different combinations of rules give different objets.

Sequence of rules for Scu production

within a certain area ~~general~~ put a point

Put another point within the area lower then first

..... etc

connect the points to make a line

rotate the line to make a cylindrical volume

connect the volume with horizontal bands to further structure the volume.

Combinations of rules can be ~~order~~ grouped to make super rule like rot & connect computer Event

would probably need a microcomputer

room is fairly dark

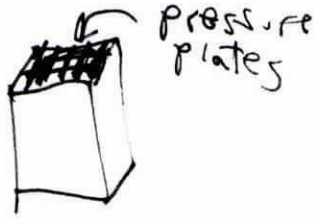
Person walks into room + strobe light goes off and Votrax says hello with an echo

~~Microphones in various parts of room~~

Pressure sensative plates & or photo cells detect approximate positon of people and set off or change a sound or light pattern

can set up an array of video screens with images That react To the peoples movements.

Set up a few pedastals with arrays of pressure sensative plates which might control The video. with a sign That says "touch me!!



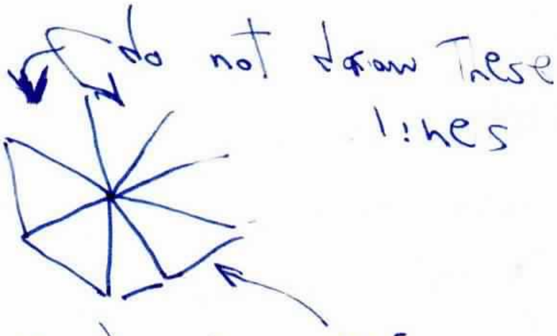
some laser light bouncing around on mirrors

lasers controled in some way.

To create a bottom and Top to sew cut outs
18 side

need to find radius

~~draw line~~



use connect To draw the segs

~~extract one seg a top and bottom connect line~~

~~from a seg~~ The find top + bottom radii

# of rotations is the # of sides

4/13

Motion

A way ~~[illegible]~~ to define speed of movement along a path.

Assuming the ~~[illegible]~~ path is 3D or 2D and linear

Need 2 control one is a steping buttom which will move you along the path a point at a time and the next is a knob which is used to define the speed (# of interpolations) from the current point to the next. This is done by listening to an auditory signal (ie a beep) the faster the beeps (controlled by a knob) the more ~~[illegible]~~ lesser # of interp. also for the sake of exactness or ~~[illegible]~~ ~~repeal~~ whatever that # should also appear on screen.

~~[illegible]~~

Using this method seem to be a good way of defining speed easily ~~[illegible]~~ and simply.

P.S. There should be two sets of points which deal with the path one is the actual path the other is a set of equally spaced ones used for the stepping and speed setting this way an object will not travel ~~[illegible]~~ to long around a curve.

Steps for speed definition

Divide the path into an equall no of points

Start at beginning point or any you want (good for editing speed)

Turn knob to select ~~[illegible]~~ # of beeps you want

Then step again. Stepping could be done by cursor positioning

however for ~~a~~ complicated Jumble'd up paths ~~-~~ a stepping button would be desired.

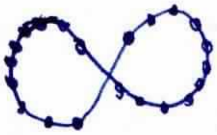
quick changes of speed may be Jerkey so an ~~averaging~~ smother program could be implemented which would make a continuous increase or decrease in speed from pt A to pt B.

Another way To avoid The Jerky speed changes would be To have an automatic stepper which would step Through The whole path quickly 5 seconds and you Just Turn knob.

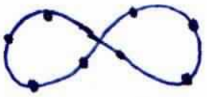
Need real Time Animation

WAIT!! Best way would be to have The cursor or a dot move along The path as what ever speed The knob is set at so changing speed is ~~autom~~ seen right away || you get what you see.

path with 2 kinds of point systems



real xyz points
which are The path



equally spaced points
for speed definition

An object can also have rotational movement while moving along The path (ie a rolling ball)

~~Need to~~ For a real Time system you could do The follow

~~as~~ Since rotation is done around a line control a lines ~~to~~ position with 3 knobs x y z or a Joystick or whatever and with a knob control speed of rotation around The line can have The same auditory beep as speed part.

where This rotation occurs on The path (if any path at all) can be controlled by stepping through Those equally ~~sp~~spaced points on path.

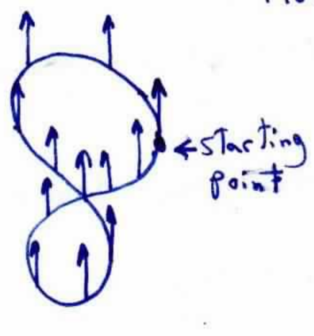
Example for whole shpeil with a real Time fancy system.

A path in 3 space is defined by using whatever method is easyest ie sonic pen. Equally spaced points are calculated ~~and~~ (shown if desired) Stepping routine is initiated and knob. which defines speed of travel along path is Turned while stepping is continuous and automatic or manul: In either case ~~there is a~~ The current position on path is shown by some symbol. moving along it.

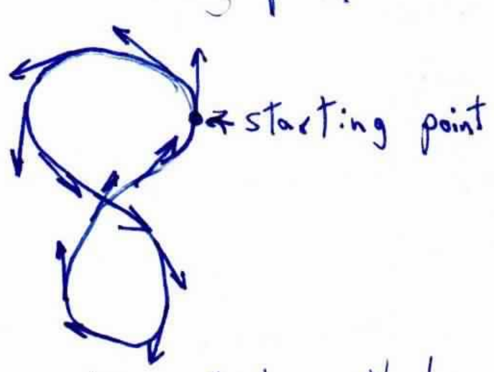
~~rotation def~~ once The speed is compleated. Rotation definition mode is started and a line of rotation appears along with object To be rotated. When dealing with path This line of rotation intersects the stepping points. And rotation speed is defined ~~and~~ with knob and then you step from point to point on path for location of This rotation.

Note: When an object moves around a path there are Two ways To do so. A single point placed somewhere in or around object ~~f~~ remains stationary relative to object and as point goes around path The object sticks with it. Or A line is defined relative to object but both points of line touch path so while moving say on a circular path same end of object always leads.

Movemen of obect along path



point attach method



line attach method

Problem - rotation can move an object away from attach point or line to path.

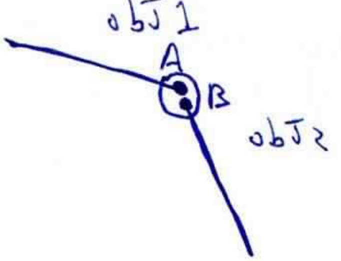
~~solution? when defining a rotation in relation to a path it is simply forbidden~~ .

Solution if a rotation move an obj from it attach point or line simply move it back // This does not cancel the rotation because this is a straight translation and the first movement was a rotation,

---

When dealing with multiple objects which have relationships to each other (ie parts of the body)

The attach point for path travel could be a point ~~on~~ on one of the objs



obj 2 has its path and speed all worked out

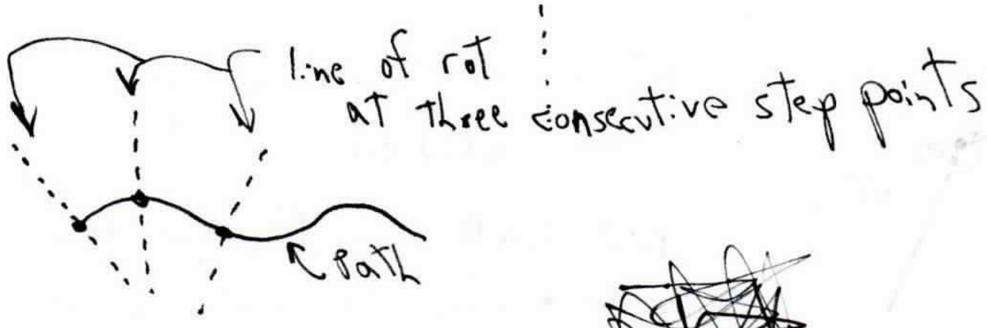
Point A & B are the same point always as far as The relationship between obj1 + obj2 are concerned

So Point B's Travel becomes the path for obj1 and A is the attach point

4/15

Back to single object manipulation

An object is moving along a path by line attach method and now I want the object to spin around in some way while moving along the path. ~~I could set up a rotation~~. The line of rot intersects the point on the path. The rotation is set up (line, speed) Then you could say do this rot. from step point 1 → 100 or 1 → 20 and 40 → 73. What happens is that the line of rot is translated and rotated so that it keep the same relative pos to ~~the~~ the step point that it is intersecting



If the rotation breaks the 2 point attached path travel Then you are automaticaly switched to a predesignated single point attach travel.

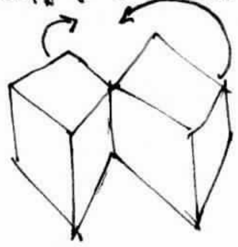
Multiple objects in motion

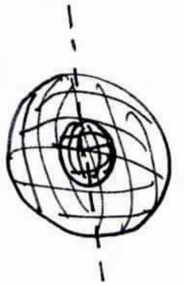
How are objects in motion related in space.

If there is no apparent relation Then simply make To complete paths and movement and display Together (Just a temporal relationship)

Two objects may be related by always having one specific point (on each of the Two objs) always Touching or a line always The same
(Two sphere rotating around a common axis)
or
Two cubes roting around a common line

→ 2 body parts (foot & calf)





Once 3 points are always in common The Two objects can now be considered one object because there is a common surface and if any separate movement is wanted one of the 3 points will break contact.

What seems to happen is that the motion of one object is dependent upon the other, or One object is a function of the other. Which is Which may be arbitrary depending on what the situation is,

This seems To be ok but what do you do when you have 100 or more or so objects moving around?

The way that They are related To each other must be found.

This seems to be related to the problem in chemistry & physics of waiting to follow the motion of all molecules in some substance (say a gas in some confined area). The Heisenberg uncertanty principle of computer graphics.

So for now any group of objects that have a complicated relationship to each other must be work out for each case. So now on to the relationship of parts of the body to each other.

To start off with there are common points of attachment. The problem is find all the constraints and defining them so the computer can examine a figure and determine whether or not a position is possible

Break the constraints down to primary secondary 3° ....
a tree structure of constraints ~~the~~ just like the attach pts.

4/16

The constraints can be stored as volume matrices agains which the body parts are compared!!

The position of the volume matrices are dependent pos of certain body part.

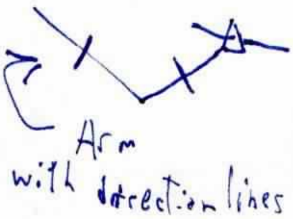
ie Volume matrix for foot constraints will always be dependent on position of calf

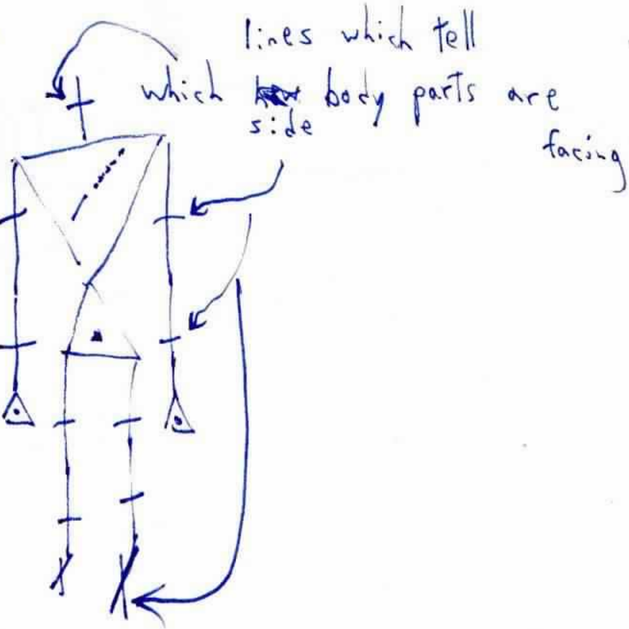
hand dependent on fore arm

The position of These volumes must always remain The same relative To those body part They are dependent on.

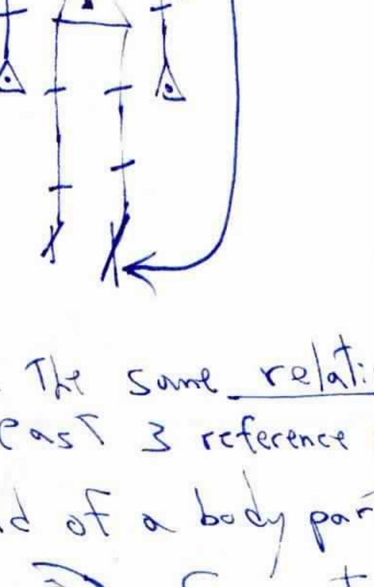
The body part can be Treated as distorted cylinders however it must be known which side is front and back.

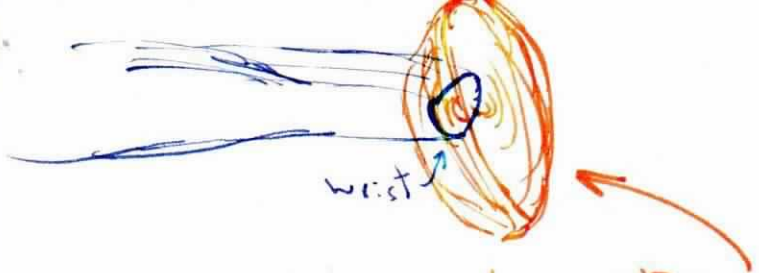
In an early phase a single line can Take place of cylinder with an intersecting line Taking care of rotation direction



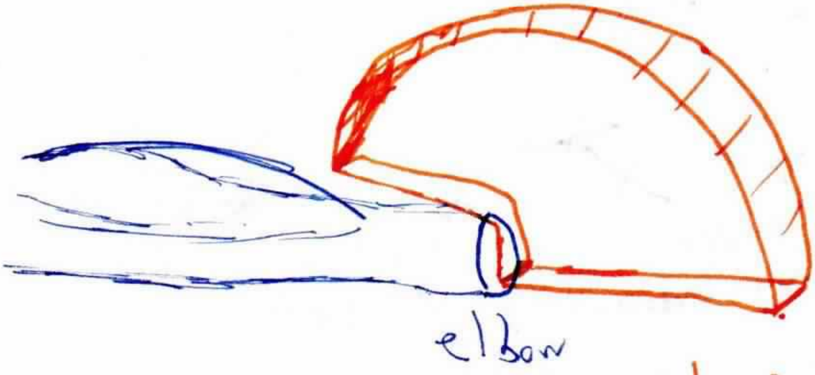


In order to keep The volume in The same relative position it must be compared against at least 3 reference points (a plane) These 3 points can be at The end of a body part and part of a cross sectional plane



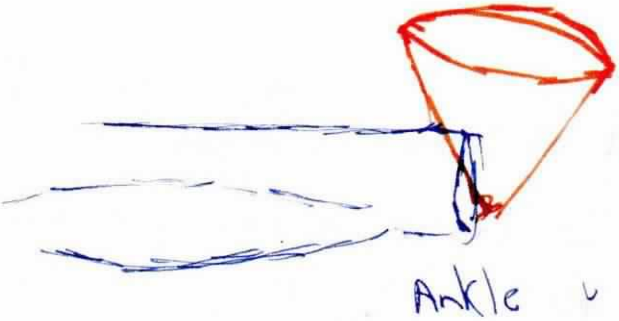


volume
of reach for hand

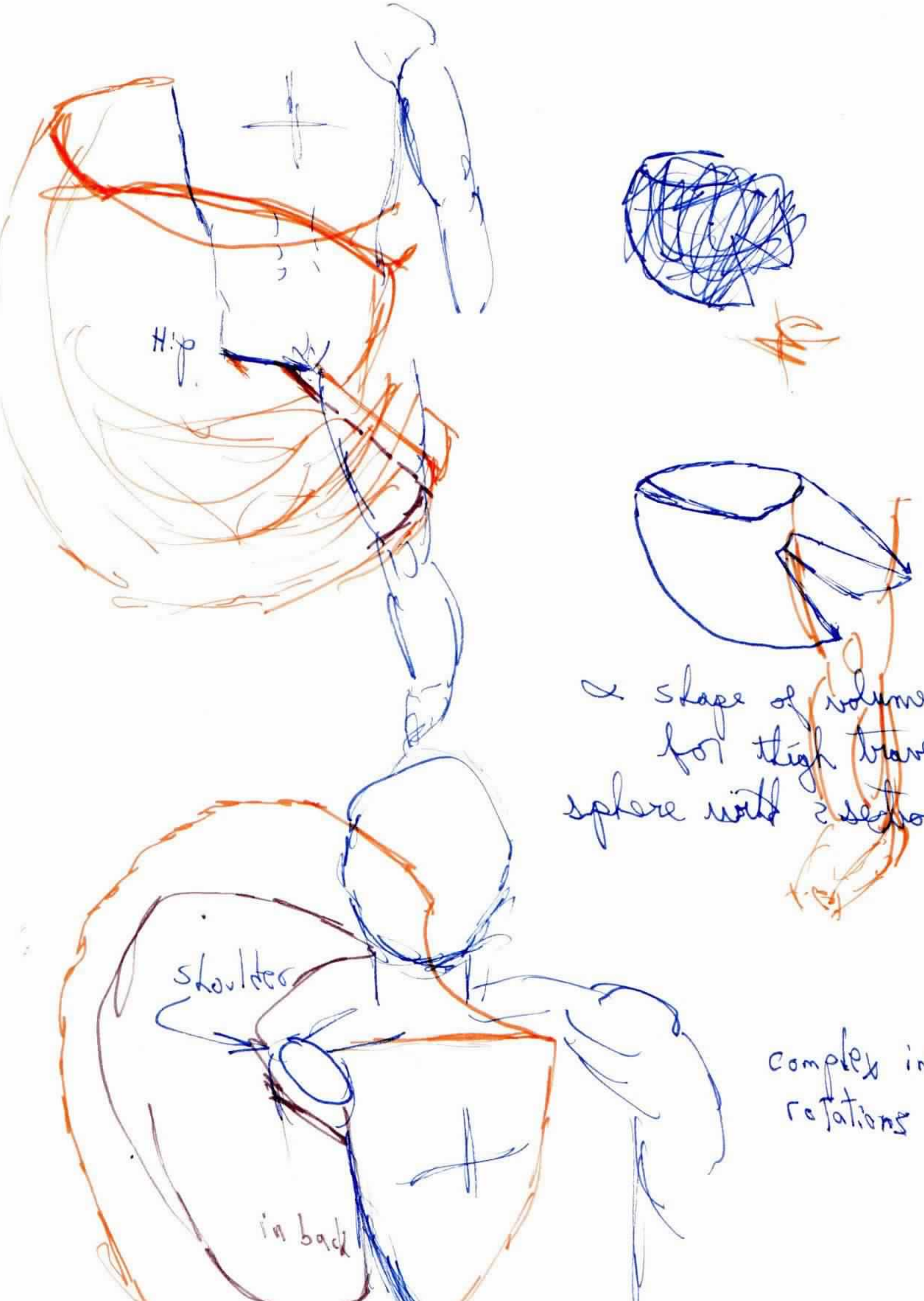


volume for forearm
only a plane of movement

internal rotation of upper arm swings plane all over



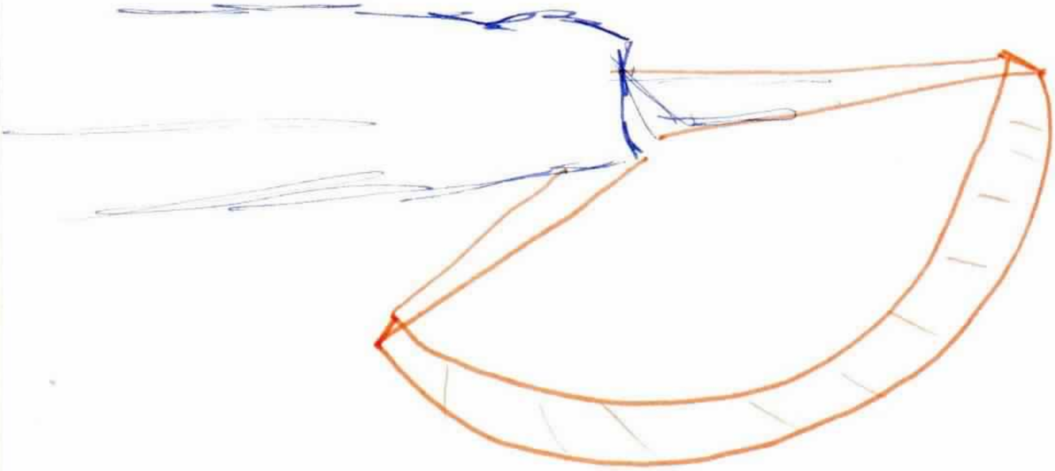
volume for foot movement



α shape of volume
for thigh travel
sphere with 2 sections cut out

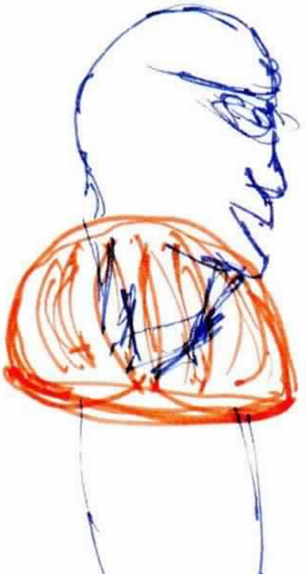
complex internal
rotations going on

knee

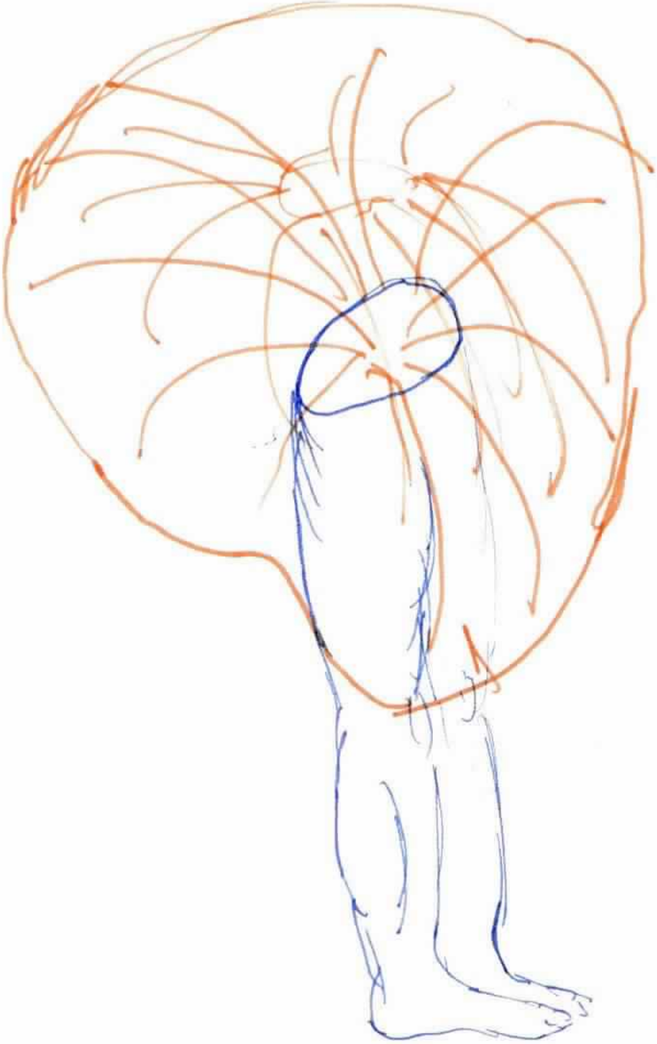


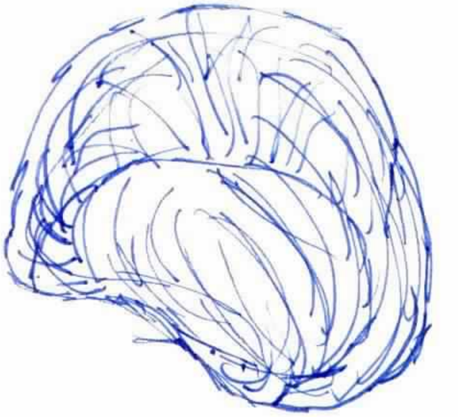
one plane of movement same as elbow calf volume

volume for head

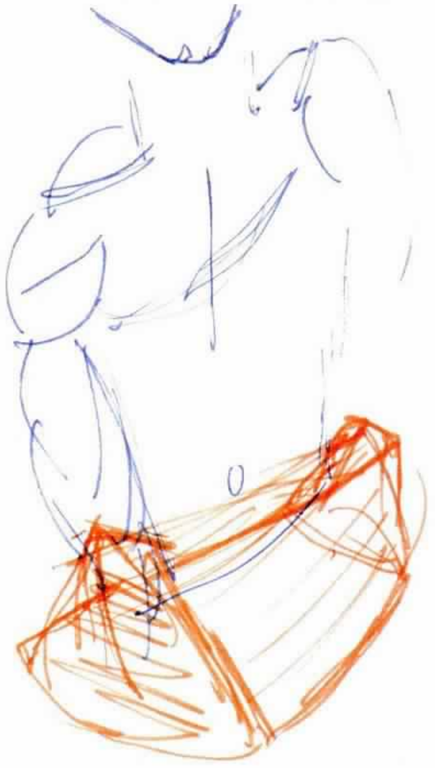


shoulder movement volume





volume shape
for spine to travel
spine is not a straight
line and is unique



Volume for
Hip Travel

ultimately all body parts absolute position become dependent on either Torso or hip position so maybe a good way to do checking for constrainsts is to divide body into upper & lower halfes. Figure out all of that then see if relation of Torso to hip is possible controlling The body with hardwar (knobs, buttons and other goodies)

Push a button and a reset value of body appears.

Select a body part or group (arms) and it becomes active.

Using light pen or Sonic pen or Rand tablet or Knobs the active part is moved all round with the computer responding with "impossible position" if move puts body part out of its volume matrix.

Positioning of body by Knobs

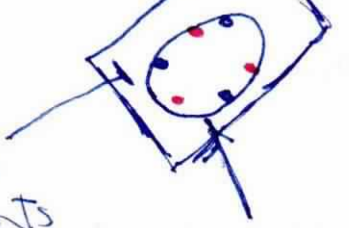
X, Y, + Z knobs for those rotations about body parts attach point

also a knob for internal rotations

constraints for internal rotations are fairly simple to check and are again dependent upon position of those body parts higher up on the tree.

The cross lines on the body parts return values of internal rotations.

4/17 Contact between two body parts must be at least 3 points for volume matrix position. So what happens when an internal rotation occurs These 3 points also rotate but must stay on the plane which ~~they~~ They define



blue dots are 3 ref points before an internal rotation

pink are 3 ref points after limb has had an internal rot.

And of course the volume matrix rotates also since thats the point of it all.

When moving a group of parts ie whole arm start constraint checking from the highest part on the Tree (The upper arm) Then move down it → (forearm hand fingers)

Although This may be the easiest or most organized way to do the constraint checking it doesn't have to be done in any order cause once the position of the 3 ref points is known that volume matrix position is known and the constraints are therefore know so ~~a~~ any order of checking constraints is possible. This may be usefull (Don't know where but who knows)

This method of checking bodily constrainst by comparing a line (body skeleton) against a volume matrix is similar to the Roberts Hidden line algorithm and so it may be time consuming although it should run considerably faster becaus I'm not checking a volume against a volume but only a line against avolume. Which all goes to show that I have no idea how fast it will run.

Probably the hardest thing to do will be to describe those constraint volumes

It would be nice to be able to do things like positioning the man by pulling sections like the hand. Pulling the hand out to the side so the torso bends over.

~~You could~~ What happens is that the part of the body which leads has set ref points and those parts attached to it are pulled as far as possible while still in their vol. mats. Wherever the part being pulled is on the tree structure of body parts you start the stretching by going in both direction of the tree until you reach top & bottom or until parts are stretched enough so that the lead body part is hooked up with the rest of the body. If the direction of the lead part is towards the left then when a choice occurs for stretching either a left or right sided body part ~~stay~~ choose the left. ie stay to the side being pulled. or towards the direction of movement.

For special cases of pulling The order of stretch may be put in by hand.

4/20 Scu$

Draw each layer separately
as in layer function

LLine ← 0 x y z
1 x y z
need to fix so it draws better

~~Bot~~ Conrot LLine

Don't deal with sections of faces
deal with lines so There is less redundancy of drawing on plotter.

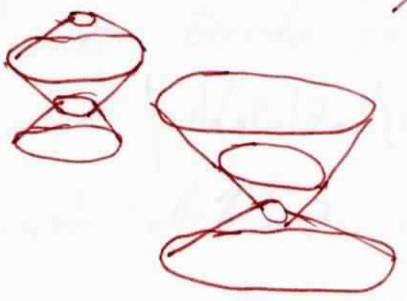
Can do an axis function on the layers and rotate Them so you see ~~them~~ Them from a top view

A Sew machine

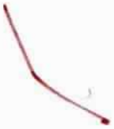
String + rings

~~won't work for~~

~~This will want To~~



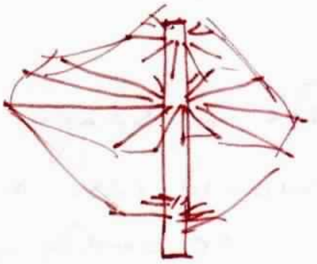
cut out styrofoam ring and cast them
They would have holes in them and
The rings could then be Threaded
with aluminum wire!!

for a shape like 
would need a knot
or something to prevent
ring from slipping down
Maybe nothing if wire Thick enough
Armature wire.

Although all of The computer documentations for the scv's will be of The same format The human interpretation and Transformation into a real life object can be extremely varied. strings/inflatables/ castings/ plastic/ This interpretation of a "fully" described object is The human element.

a center supporting rod with projections out of it

can be connected with string or wire



Do all kind of different distortions To The scv and compare Them. Maybe They all have a "computer" look to Them.

Make so many scv's until it is both amazing and nauseating. These are two of the characteristics of computers in society. They can be amazing or really screwed up.

4/25 Scus

~~charvec ← 'oneTwoThrforfiv'~~

One
two
Thr
fou
fiv
six
sev
eig
nin
Ten
zer

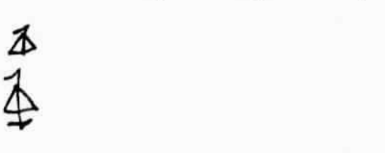
Need To create graphic numbers

input is ~~##~~ #'s output is drawings.

```
∇ Graphnum ~~[illegible]~~ Nos;
⎕IO ← 0
~~Num ← 0~~

DrawGNum ← GNum cat (ΔX, 100, 0) Move ~~[illegible]~~Num

:

~~countAt~~
~~Nos[c~~

~~[illegible]~~ Chars ← ⍕ Inp

⍎ Chars[count]
```

One symbol for each drawing

· G Chars ←



```
∇ NumDraw NOS;Limit;GNum;count;NumChars;ΔX;Num

    ⎕IO ← 0

    Limit ← ρ⍕NOS
                                                    1 2 3 4 5 ....
    ~~ΔX ← count×50~~

    GChars ← 'Δ⍋ΔΔΔΔΔΔΔΔ'
    GNum ← 0 4ρ0
    count ← 0

    NumChars ← ⍕Inp

Guts:    Num ← ⍎GChars[~~count~~][⍎NumChars[count]]
         ΔX ← count×50
   Draw  GNum ← GNum cat (ΔX,100,0) Move ⍎Num

       →(Limit ≥ count ← count+1)/Guts

       ⎕IO ← 1

                                                    Inp ← 15746
                                                 NumChars '15746'

                                                  ⍎NumChars[count]

                                                     5
                                       N


 Num ← ⍎GChars[⍎NumChars[count]]
```

Write a Master program To do all Kinds of Things with scus. Make it very interactive.

At the start you can:

Make a scu

Make a blade

4
8)350
32
30

Do a combination of layers of different scus (start that process)

combine Scus

Document any process

Flaten

Shade

A very specific hidden lines routine Just for Scvs

~~If the r/OBJ[;3] is ≥ 390~~

~~Then only rotate half so you get the front half and thats all.~~

~~If there are parts < 390 got problems~~

center the scv and draw only the front half and dats all.

To center

```
x ← r/obJ[;2] − L/obJ[;2]
y ← r/obJ[;3] − L/obJ[;3]
```

```
x  (512 − (x÷2))
y  390 − (y÷2)
z  0
put obJ
```

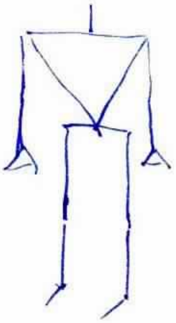
```
Draw. obJ ← ((512 − (x÷2)), (390 − (y÷2)), 0) put obJ
```

Generated Covers Top + Bottom

Given a radius of r

→ Create

What I have now in APL for Dance



Commands Position → Rotate x y z

View → Top Bottom Right Left front

Help → Bp + com List

Name → Names a position

Stop → Guess what!

.04 Anima is a slow moving figure

Need an extraction routine which extract body parts from completed positions

order of concatenation in Attach program must stay the same for this to work

~~∇ Pstt Parm ; count; Lim~~
~~Temp ← 0 4 ⍴ 0~~
[*] ~~Lim ← ⍴ Parm~~
~~count ← 1~~
~~Guts: Temp ← Temp cat Body [parm[count] ;]~~
~~→ (Lim ≥ count ← count + 1) / Guts~~

need list of Bpnames and another set of Pointers which point to where the parts are in the finished Position.

Bpname From Posname

LJA From Da1

scan down Bplist to get to right level of pointer list to get to right sections of complete Body matrix

BPNames can be used Pointers can be called Pospoint

~~∇ from Posn~~
~~[illegible] from Posn ;~~

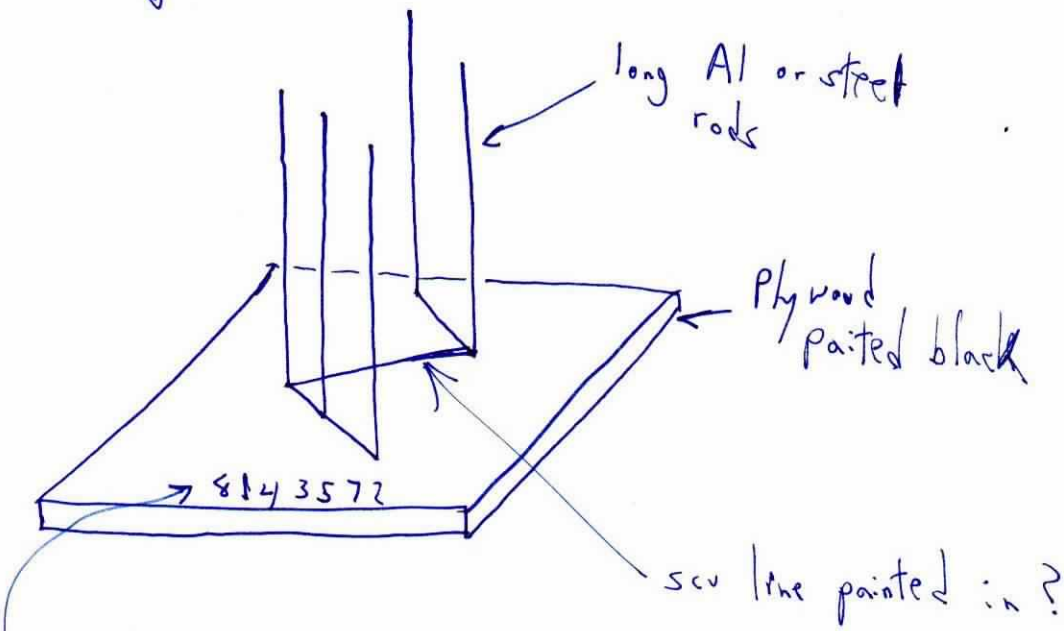
→ (Bp = Bpname[count;]) / StSeen

Ass : ⍎ Bp ← Posname[PosPoi[count; over] ;]

→(OLim≥Over ← over+1)/Ass

sculplure

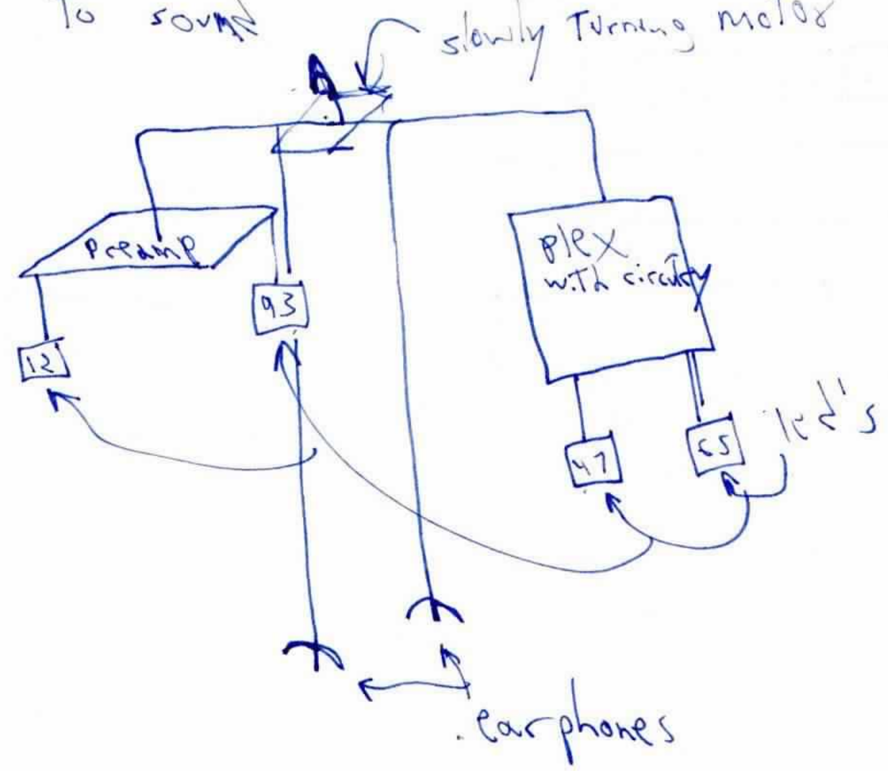
Take the randon generated scv lines and



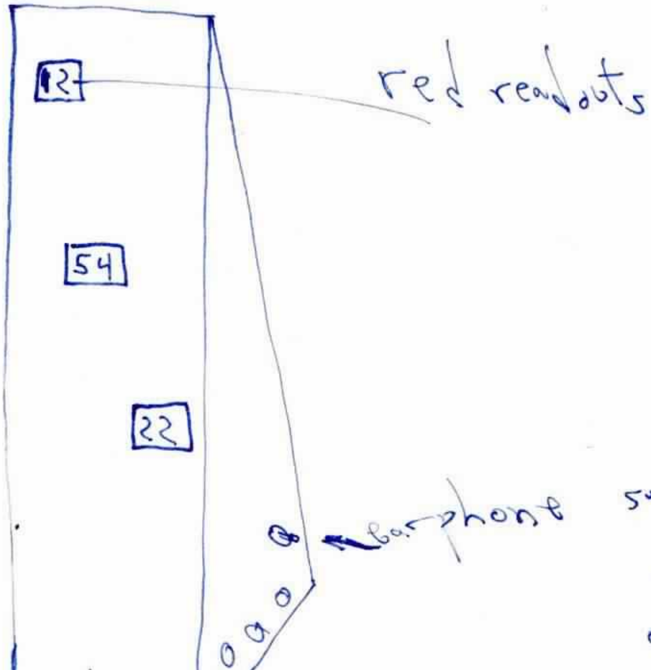
plus a paper of documentation

Get stencil of Numbers and paint the URL on

Trace points from original so as not to destroy original drawings make an electronic mobile That reacts To sound



clock

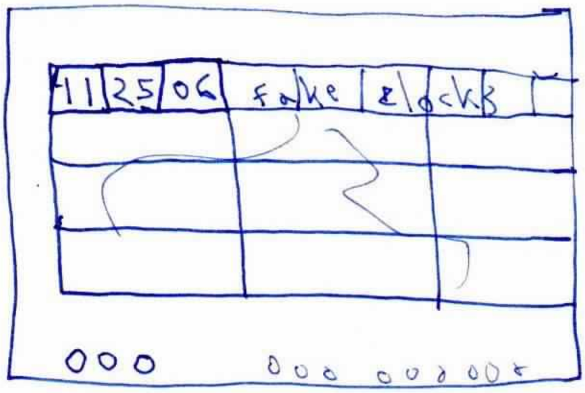


month
day
date

swith to month day date
can't be Too sensative
or else always be switching
back & forth

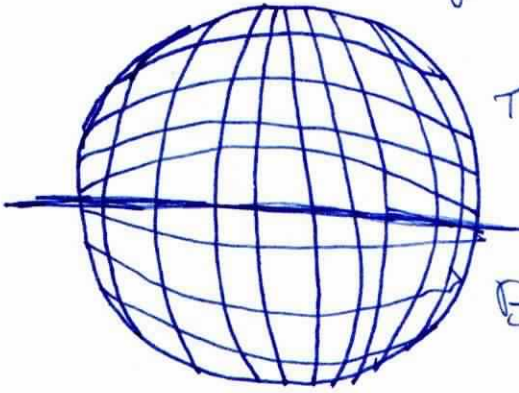
clock in Canvas

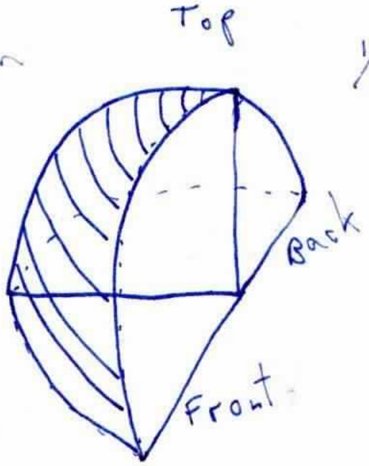
5/9



Comp. Dance

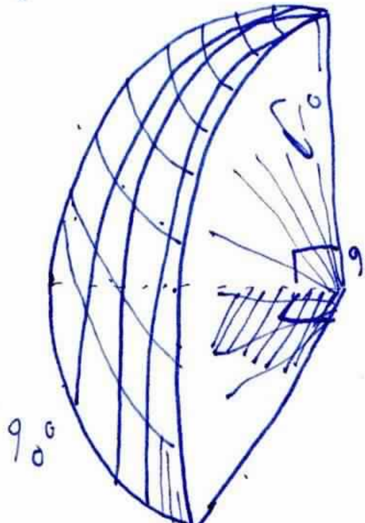
Position The bp's using a sphere as reference

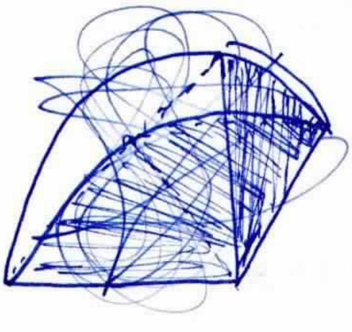




1/4 of whole sphere

1/8 of whole sphere

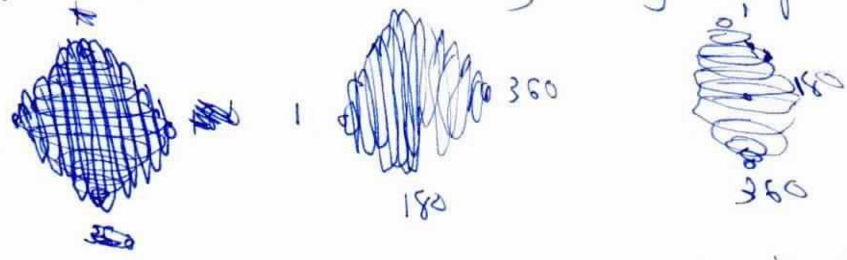




1 - 90 in hor

1 - 90 in vert

sphere is series of ring 1 degree apart



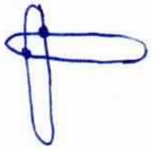
Top → Bottom 1 → 360

Left → Right 1 → 360

front or back intersection

Example

90
90



front or back

also need an internal rot which would be around ~~axis~~ line of dp itself.

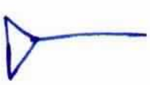
command: Position

what is the BP

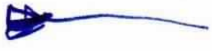
LA

~~vert & Horz~~ v front or back & int rot
Hor vert
180 180 F 0

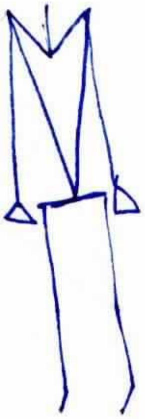
1 180 F



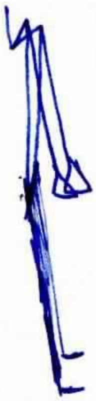
1 180 F 90



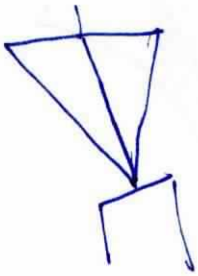
spine now sticks with shoulders

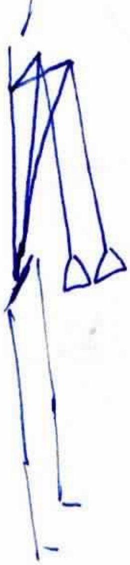


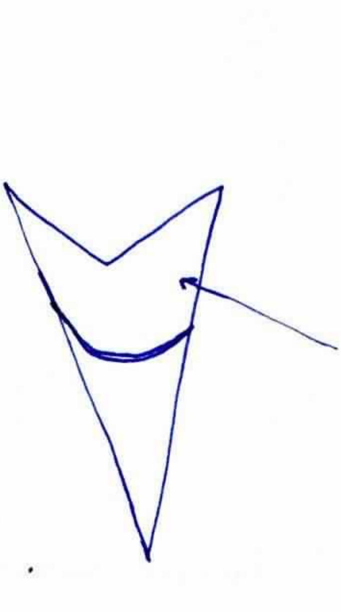
problem →



need a center line for side views
and maybe also front

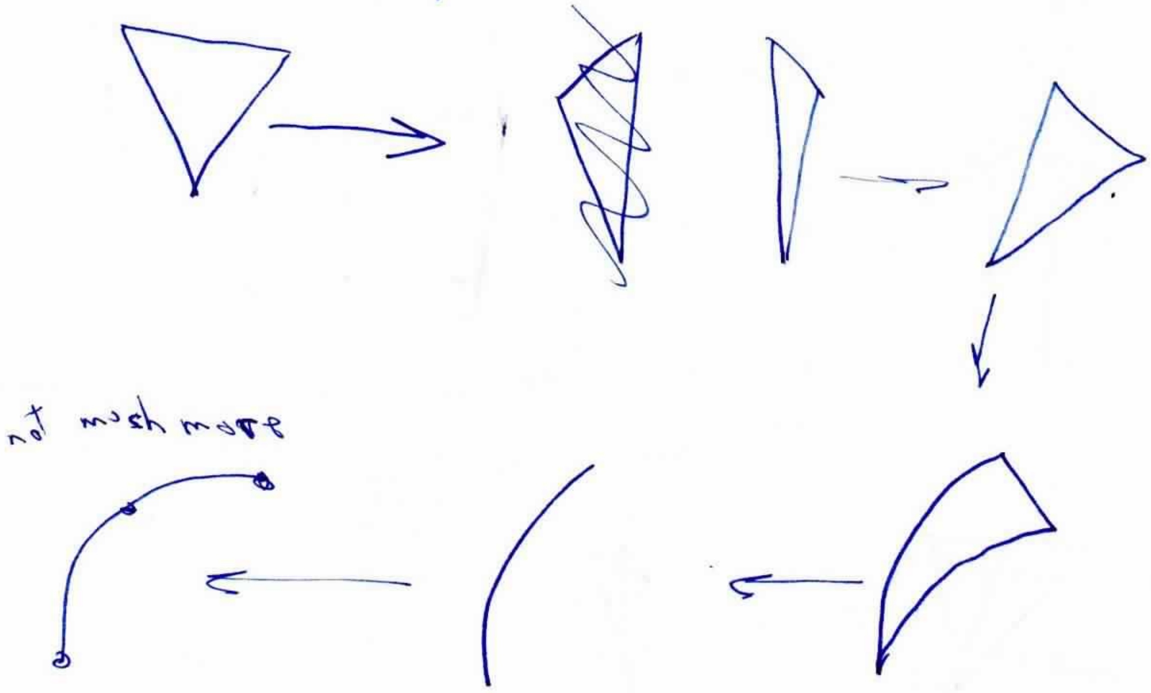




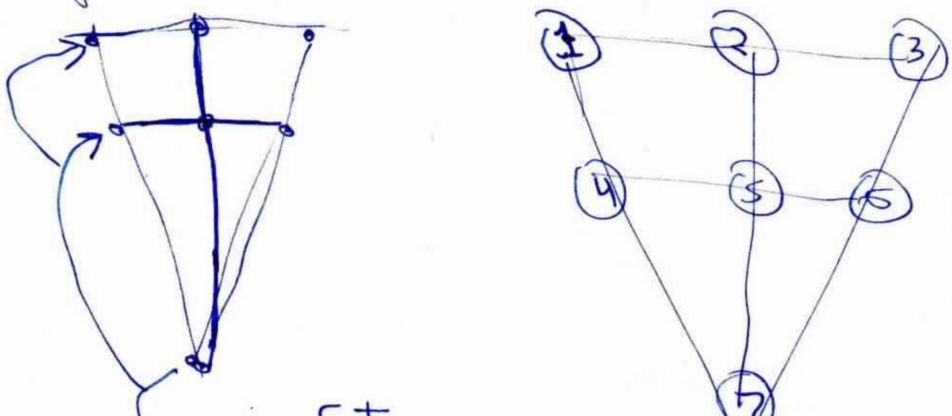


This section
would look nice with
curve fitting

let's bend that spine



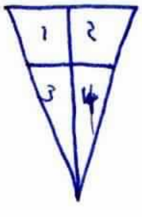
spine based on



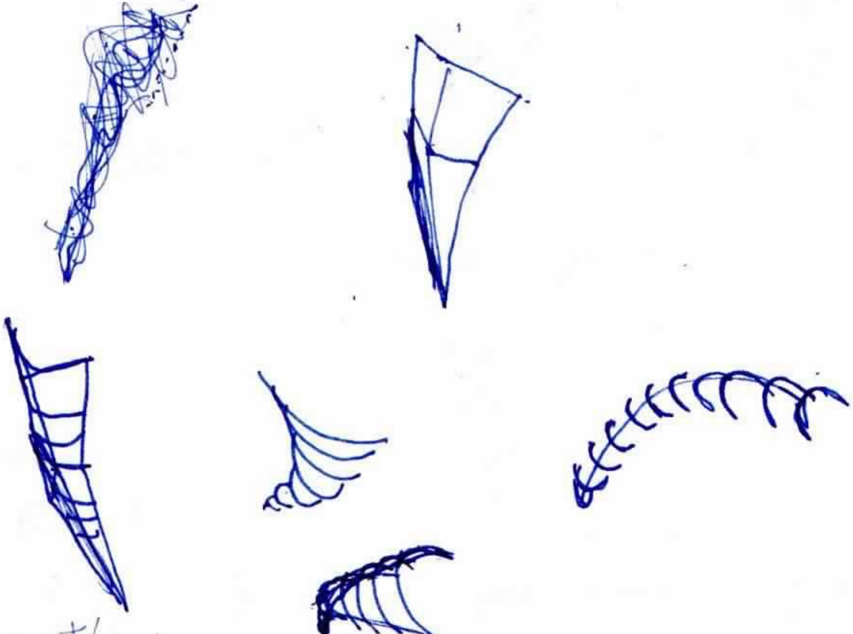
curve fit 2 5 + 7 forward & backward bend

curve fit 1 4 7 Then 3 6 7 for Twist

put the lines Together linear interpolate from 147 to 257 to 367 and you have a surface which is Torso only a surface not a volume but damn good



Four sections with curve fitting would look pretty reasonable

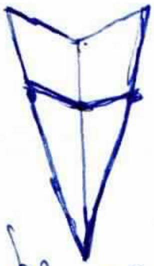


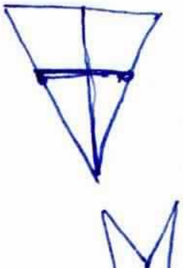
dividing up the Torso into more sections would make it better but how do you control it

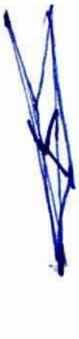
Need a model of Torso action and then a control system

Try and keep it simple.

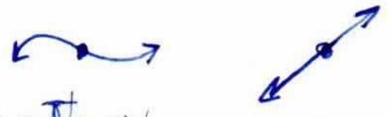
should be able to stick the crossbar in now without any curve fitting (maybe?)

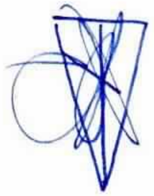


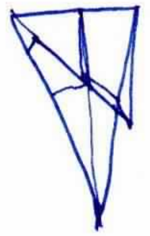


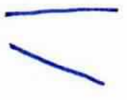


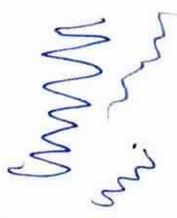
Bend And Twist factors be needed

Twist would  separte the crossbars from the side lines

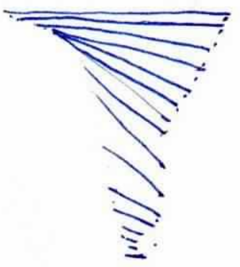




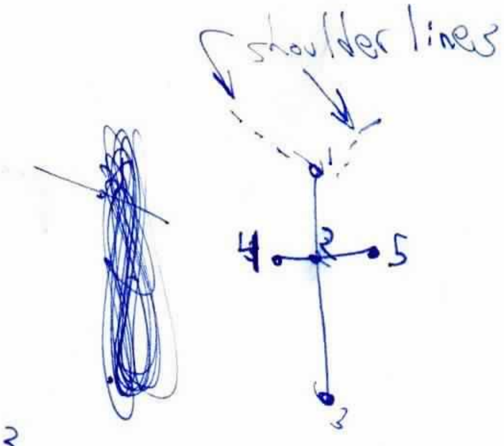




linear interp from shoulders to crossbar To base



How o do Bend ~~with ben~~



Bend up forward is a rot of line 1-2

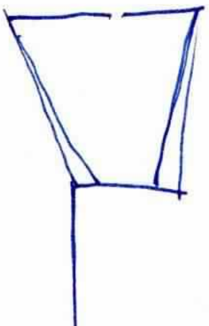
Bend bot is a rot of 2-3

Twist is a rot of 4-5 around 2

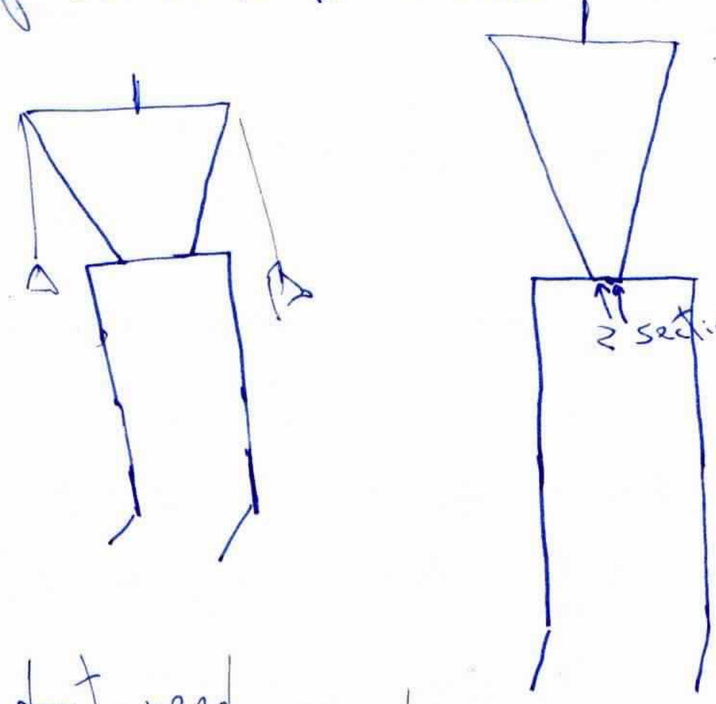
maybe for better control separate 4-2 and 2-5 so ther would be 4 segments for Torso positioning with interp from shoulder lines to crossbars To base

shoulder lines ~~not indep~~ are controlled somewhere else but become parts of torso

~~base should be~~



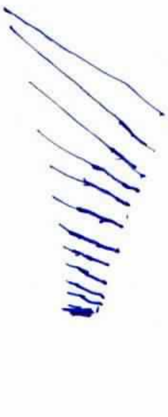
Base should be extended To be a shorter version of hip so it to tike shoulder lines moves



for base 2 sections ~~Bal~~ That follow movement of hips

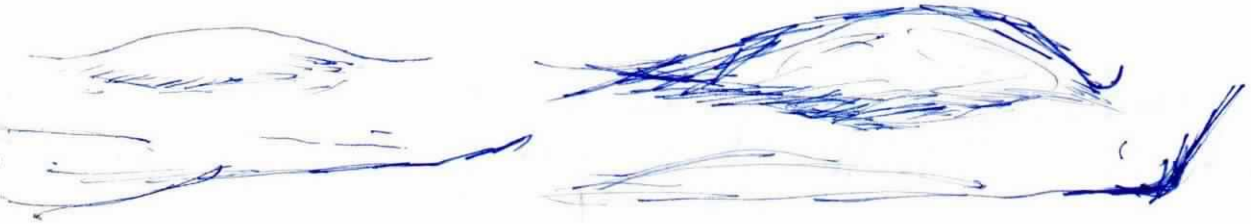
~~Maybe dont need cross bars if you interp from shoulders To hips~~

Keep cross bars



5/10 Muscles and ~~Dance~~ computer Man

can changes in muscles during flexing be represented by a linear Transformation



need a study on the gross anatomical changes of muscles

Muscles looks would then be dependent on position of skeleton (stick figure)

The transformation of one surface into another and some spots would probably overlap.

maybe there exist a few basic (primary) surfaces which could be put together to make secondary surfaces that would look like flesh and muscle. A moduler approach to the problem Now the only problem is figuring out what these sections look like. Remember that they can be scaled in x y & z separately.

start with cylinder and distort sections Then extract needed parts.

Photographs with grids ~~or~~ on surface of body
like a very accurate cross contour drawing
but in both horizontal and vertical directions.

Mirror image patches for left and right halves of body

Since most muscles ~~like~~ work in pairs one muscle change might simply be the reverse change of its pair which is dependent on skeletal position.

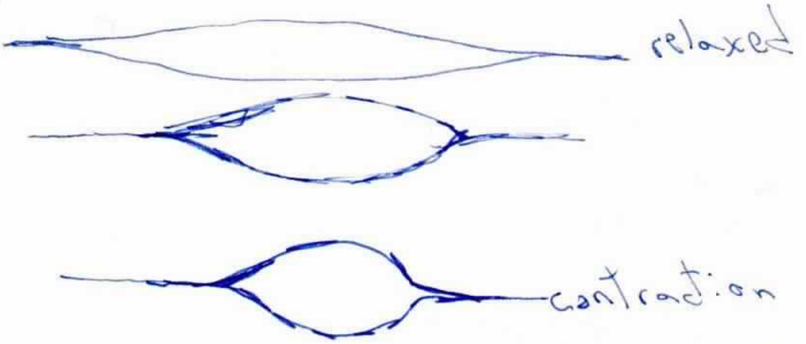
Must work out full relationship between skeleton and muscles

Maybe muscle changes can be worked out so that The muscle moves The skeleton like the real body!

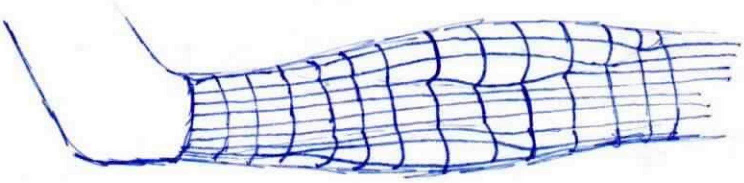
Constraints are a problem and would be needed probably

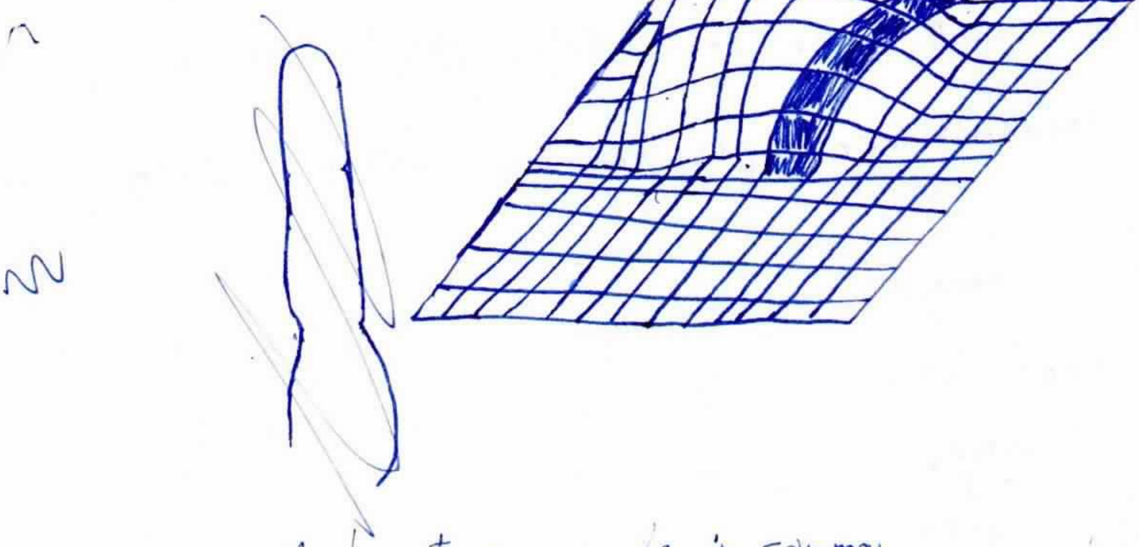
If changing muscle shapes is easily controlled you don't want a user To have To know all the muscles if any.

Human	Brain	nervous system	muscle	skeleton
computer simulation	User	program	surface patches	stick figure



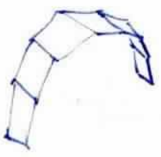
muscles get shorter and fatter on contraction.





Maybe strips can be in common and bending strips contracts muscle

bending a strip would be similar To bending spine.

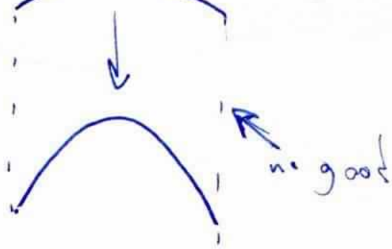


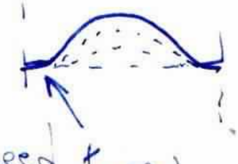
Top strip of muscle
or strip of maximum
bend.

create a hierarchy of strips
and control ~~for~~ muscle action
by controlling strip

This strip inter affects a certain (most or half or all) of rest of muscle | Some muscles may need more then one leading strip and so action would be essentially 2 sub muscles working which would really make up one complex muscle.

How does this lead strip affect surrounding areas.

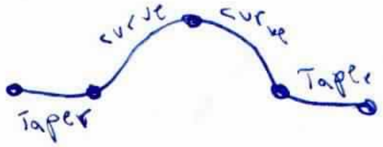




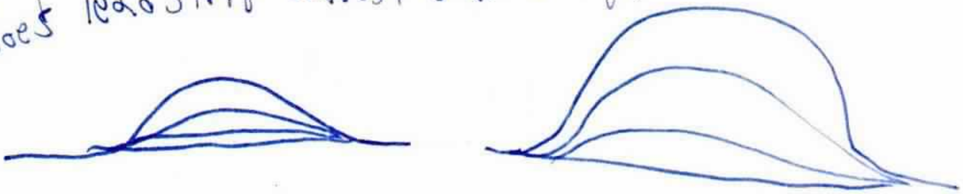
curve of muscle can be based on 3 at most 5 points

The old curve fitting shtick

3 points to define curve
2 for taper



How does lead strip adjust other strips?



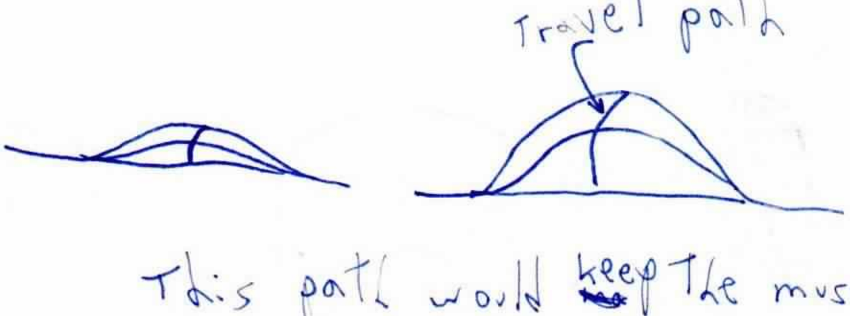
~~lead strip affects ## of interpolations~~

if you bend lead strip and keep number of interpolations to bottom constant Then it will look like stretching

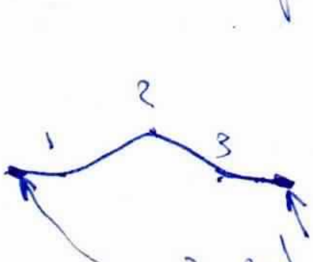
but it would flatten out sides.

adjust strip according to hierarcy of same kind

if you know maximum and minimum positions of muscle Just let strip be evenly spaced along a Travel path!!



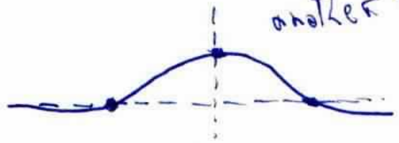
This path would keep The muscle round.



2 end points fixed To parts of skeleton

Move points ! 1 3 for Tapering adjustment
2 for extent of contraction

another travel path for control points



~~moving control points and~~

moving skeleton ~~controls~~ affects control points of muscle by moving them along their travel paths

moving control points adjusts stripes along their travel path

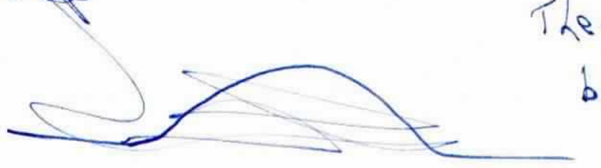
moving strips ~~[illegible]~~ changes shape of muscle which is what you want!!

Difference between one muscle and next would be locations of control point travel paths and strip travel paths.

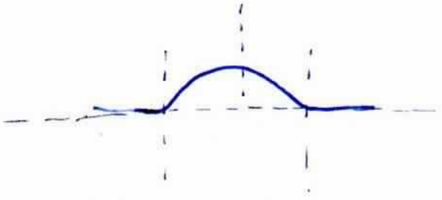
So changing travel paths changes structure of muscle

So maybe only one ~~[illegible]~~ muscle is needed and what is stored are the different travel paths.

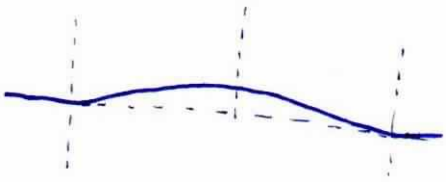
Crops



There would be a top and bottom half of each muscle with 2 lead strips however one ~~[illegible]~~ lead would control the other.

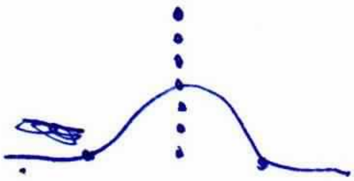


fat muscle

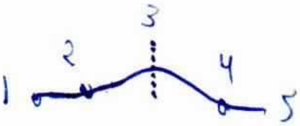


Thin muscle

Travel paths must have limits but no of points for Travel can remain constant. Spacing is what is varied from muscle To muscle



4 up on path for fat muscle

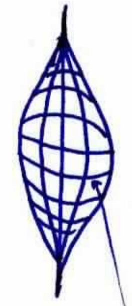


4 up on path for Thin one

distance from 2 to 3 and 3 to 4 varies..... The farther they are the skinnyier

also 2-3 3-4 distance not necessarily The same. it varies with structure

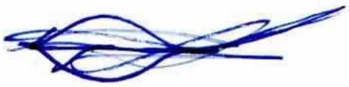
minimum and maximum contraction values would be locations of control points on Their travel paths.

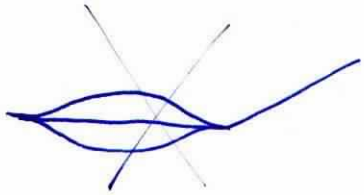


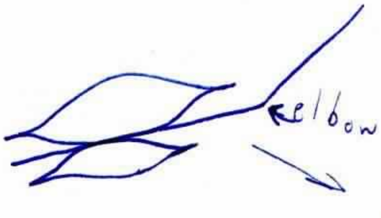
hor cross lines would be put in and connect

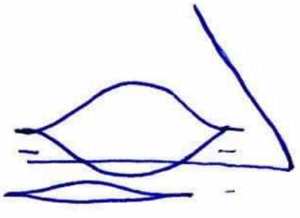
Obviously The first Thing To do is get a curve fitting routine.

way That skeletal angles affect muscles .... (control point positions on travel paths) would have To be different for all or most muscles









need a skin covering all of This Just does muscles and The dancer would look kind of strange with raw muscles especially without a full complete set of muscles.

Skin could be ~~con~~ lines which go from sides of muscle To muscle.

5/11

The muscles are the movement constraints.

How →

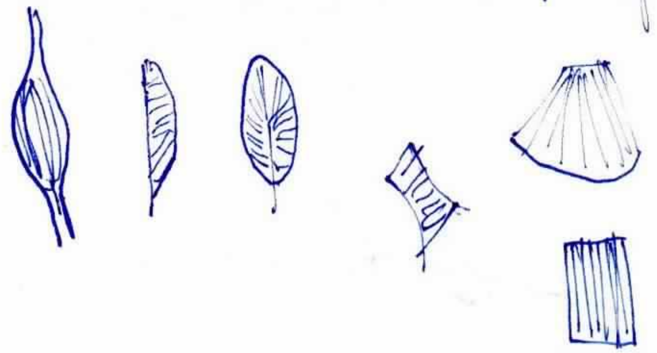
You position the skeleton

The skeleton affects the muscles by changing.
control points.

If control point has to go past its travel path
Then its going past its physical body constraints
and you can do it so you get same message.

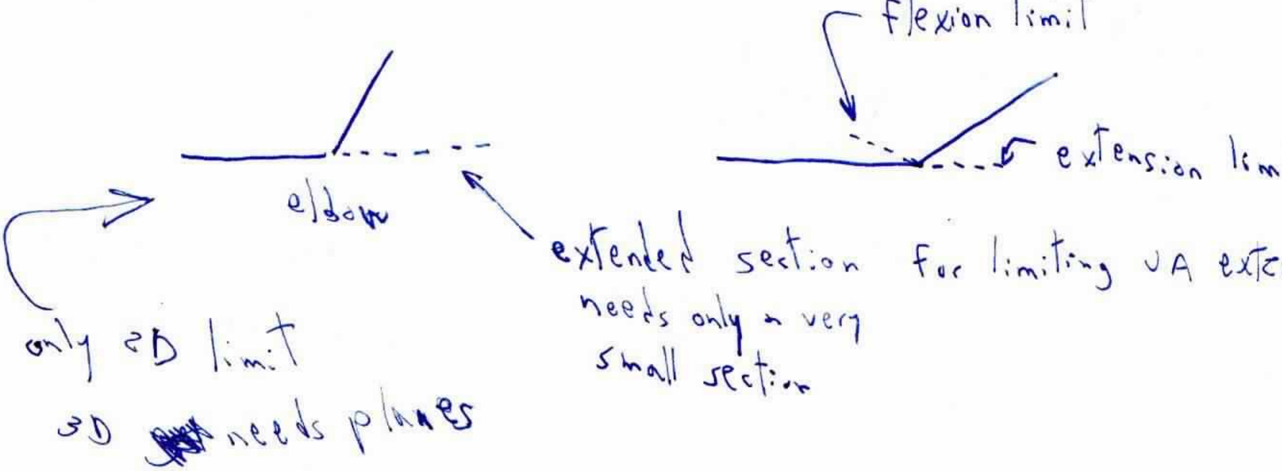
So The length of Travel paths <u>are</u> the constraints.

Different Kinds of muscles shapes



Basically only 2 kinds flat and cylinderish.

Constraints could also be handeled by adding shticks to skeleton which on contaret with parts would signal a limit to movement


flexion limit
extension limit
elbow
extended section for limiting UA extension.
needs only a very small section
only 2D limit

3D needs planes or surfaces

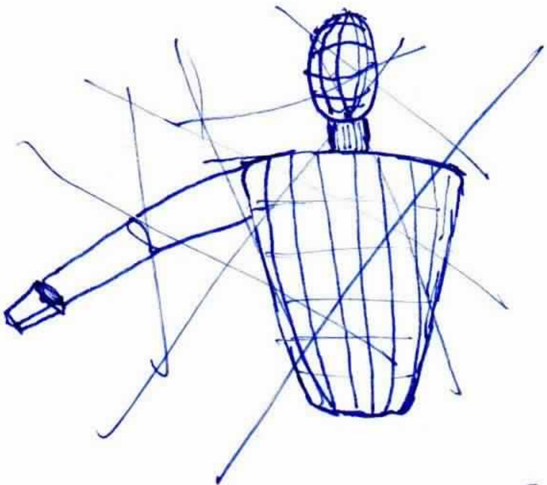
This seems to be leading to constraing shells and back to the old volume matrix!

~~can constraint be fully handle~~

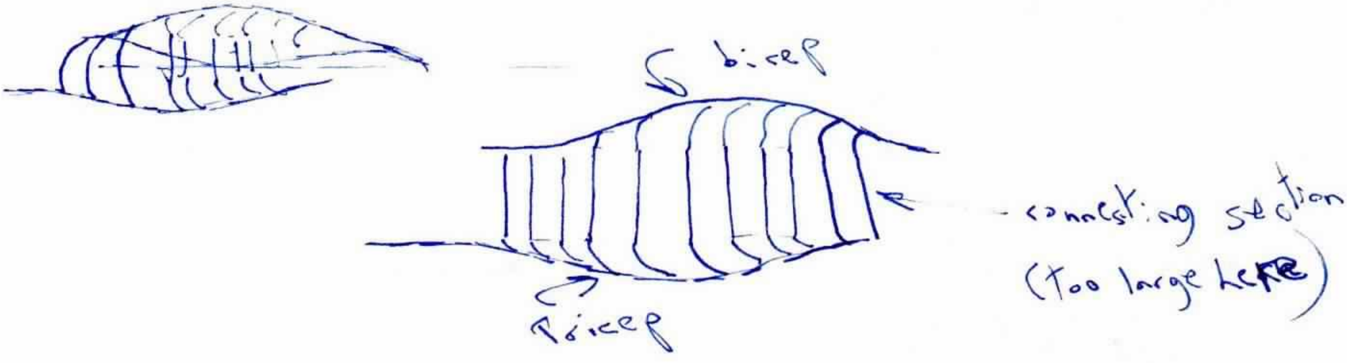
~~Would be nice to have some constraints befor.~~

Obviously won't be able to have all the subtleties of all muscles exactly worked out. yet.
So have to be able to selecte those muscles which are most prominante and put in those limitations which are most important & obvious.

5/12

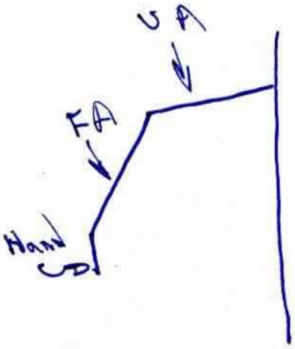


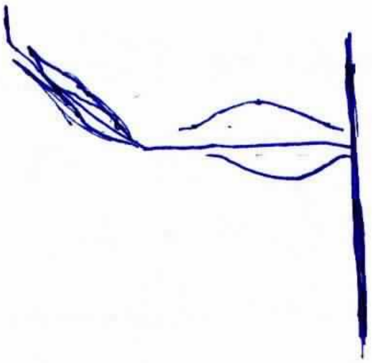
Take shape of body by connecting muscles To forms an active moving muscular Skin



Try To get a prototype arm working with muscles before attempting the Whole shtick.

reference line instead of Torso





5/22

The statement that a painting is a flat surface with various colors on it was the whole philosophical basis for "Modern Art" up until Pop Art & Rauschenberg. It totally ignores the idea the human perception is involved in all our senses. It is the perception of art which matters not whether or not it is flat or deep or colored or.....

The period of Modern Art Manet → Abstract Expessionism was needed to free artists from previous restrictions. Now everything can be looked at as possibly being "Art."

But what now, after Rauschenberg? & where does computer art fit in. Take an aspect of society put it through a computer and get some art out of it.

computer can give you
→ Quantity
→ Variation
→ Description or Documentation

where are these things found in society

Advertising.....Quantity

↑ Pop art exploited this

Information and the quantity of it
knowledge
being produced today is much greater and faster than ever.

computers and Their relation To information.

catagorize

order

Process ... Information processing

— — — — —

Politics and bureaucratic information

↓

~~comp~~ mismanaggement of mass quantities of infor

Get the computer To mismanagge info.

6/2

The relationship between Two bodies changes so how do you describe this? A body is a collection of object with a static relationship ie only attach points always in contact.

Need a dynamic relationship between attach points ie (2 bodies)

electric charges

+ —

full charge of opposits = Total attaching

no charge is neutral

Full charge of alikes = maximum repulsion

give attach points charges and values of charges or Just give values of -1 To +1

these values could be dynamically changed thus dynamically changing The relationship between Two bodies

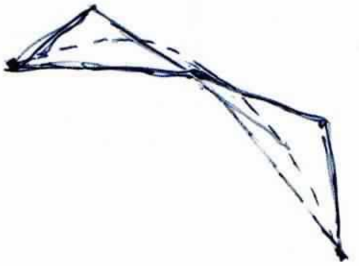
Problem: what does attraction and repulsion (partial) means graphically and exactly.

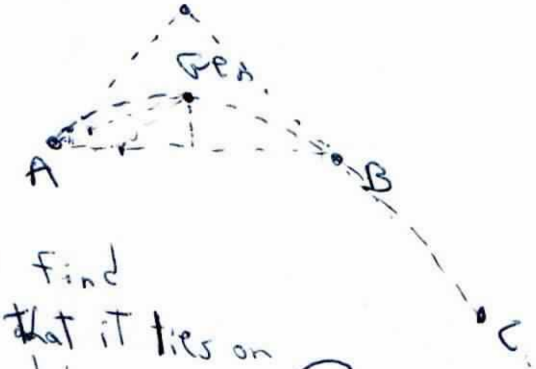
distances between attach points

it might be as hard To dynamically control These charge values as it would be To play Through positions.

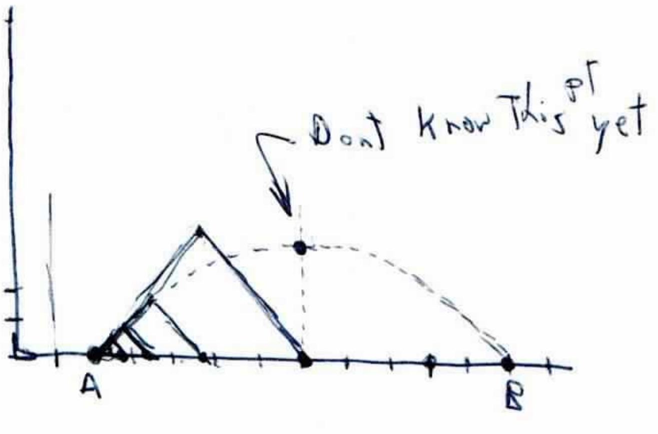
5/5

Curve fitting routine is needed for muscles To work.





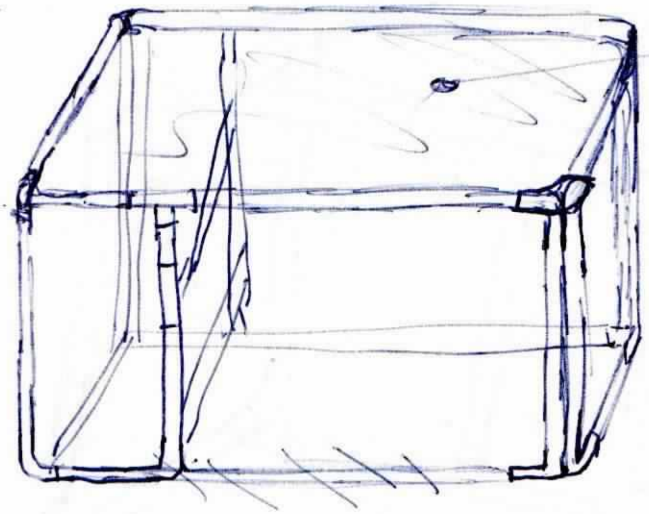
Need To find G such that it lies on a reasonable curve for $\widehat{ABC}$



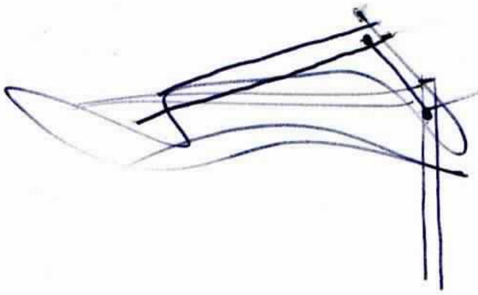
6/9

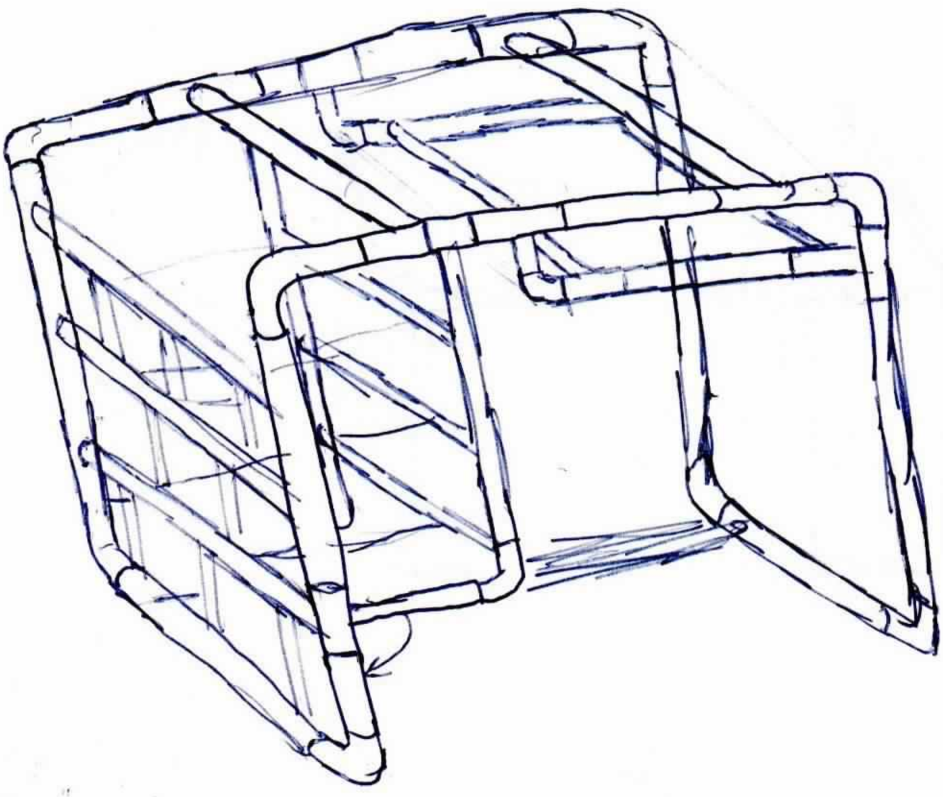
Pour acrylic out on ground
and inbed rock or metal in shapes
of sew lines.

6/20 Make Desk out of PVC pipe
would be light & strong



Plex Top

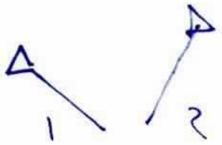




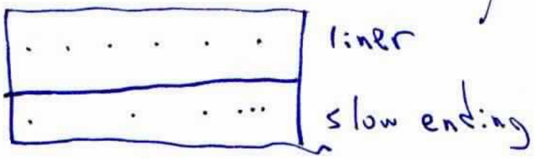
6/25 Dance

need To make BoJ more manipulable.
problems in creating position
~~can't~~ get directions for + & - degs mixed up a lot
time wasted in waiting to see other vp's

quality of movement



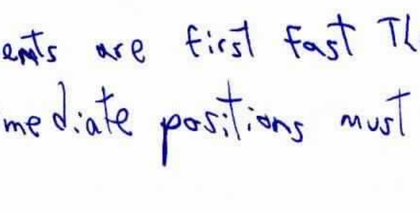
movement from 1 to 2 now is with a linear interpolation some movents are first fast Then slow so spacing of intermediate positions must be controlled in a more refined way



Sequences of Dialogue

Pos
what BP

LH
x y z°
~~0 0 Back45~~
45 Back 0 0



Back is translated into either + or - sign.

Command:

Pos

what BP

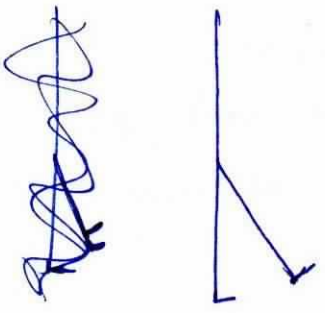
RL

what are x y z angles

~~0 0~~

Forwards 0 0

side view



need capability of working on a single bp for many angle changes

Com:

Pos

what BP

LL

Angles

0 0 20

Angles

0 0 -30

Angles

0 0 -10

Angles

NC

NC means new comand

Command: Pos

what BP

RUA

Angles

0 0 10

Angles

0 0 10

Ang~~les~~

NBP

what is The BP

LUA

Angles

0 0 10

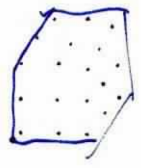
NBP means New Body parts and keeps you in positioning mode.

6/28 Dance

Need to define and classify movement in such a form as to be translatable into Labanotation symbols.

Laban deals with movement in a defined space a cube about the movee

There are 27 points of reference



a movement can be defined as going toward one of these reference points. // Does Laban do this?? //

Think so

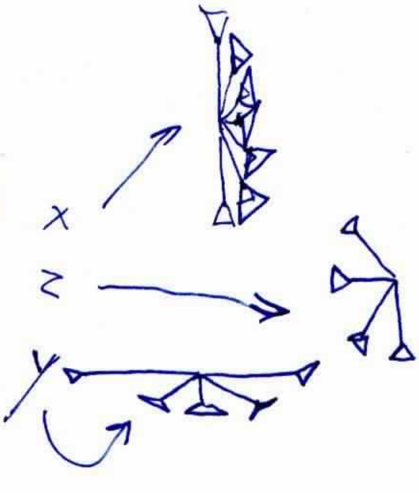
every bp would have its own associated space within to move

Need some common syntax of movements

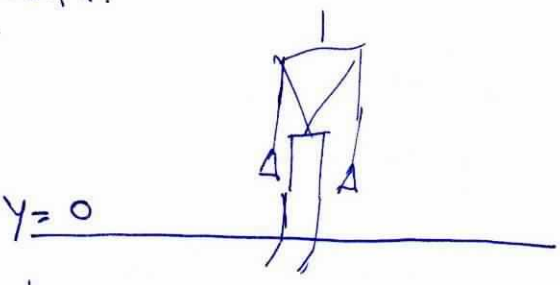
Twist
Lift
Lower

Actions of movement

Forward & Back	rot around	x
Vertical side	rot "	z
Horizontal side	"	" y



need an awareness of ground.



when any part of body is < 0
move body up

special condition for foot
when it levels out lower body

for Jumps This awareness of ground must be turned off

Absolute Scale

$oldsize \times X = size wanted$

$100 \times X = 200$

$X = 2$

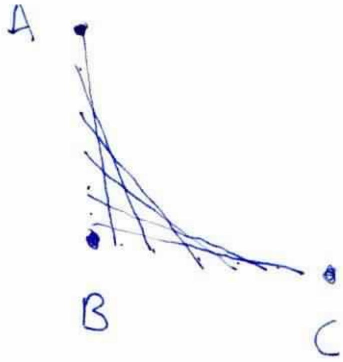
$$X = \frac{sizewanted}{oldsize}$$

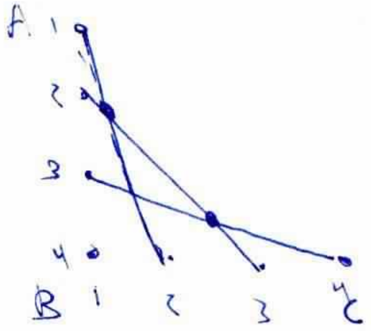
6/27

```
        ~~∇A~~

        ~~∇ Parm~~ As

        ∇ R← Parm Ascale Pic ; X          ; count ; sfac

   1       ~~∇A~~ count ← 1
fac: sfac← Parm[count] ÷ ( ⌈/Pic[; count ~~Parm~~+1] − ⌊/Pic[; count ~~Parm~~+1] )
          →(count>1)/GetB
            A ← sfac x
        ~~→(3≥ count← count+1)/ fac~~


        count ← 1

        ~~sfac ←~~

Guts: sizew ← Parm[count]

        oldsz ← (⌈/Pic[;count+1] − ⌊/Pic[;count+1])

        ~~sfac ← sizew~~
          sfac ← sizew ÷ oldsize

        1  → (count>1)/Get~~X~~     → (count=1, count=2, count=3)/Getx, Gety
                                                                        , Getz

          ~~Pic ← (sfac×oldsize, 1, 1) Scale Pic~~
              ~~As~~
      Getx    X← sfac × oldsz

              ~~→Guts~~ → Guts
      Gety:   y ← sfac × oldsz

                → Guts
      Getz:  z ← sfac × oldsz
R ← ~~A~~ (X y z) Scale Pic
```

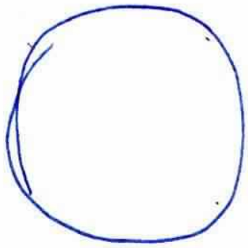
```
∇R ← Parm ASCALE Pic;x;y;z;sfac;count;oldsize

     count ← 1

   Guts: sfac ← Parm[count] ÷ oldsize ← ((⌈/Pic[;count+1] − ⌊/Pic[;count+1]))
 → (count=1, count=2, count=3)/GetX, GetY, Getz
 Getx: x ← sfac × oldsize
          → Guts
 Gety: y ← sfac × oldsize
          → Guts
 Getz: z ← sfac × oldsize
  R ← (x, y, z) Scale Pic
```

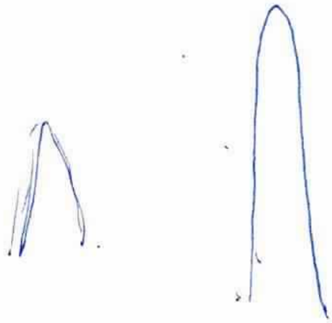
Curve fitting

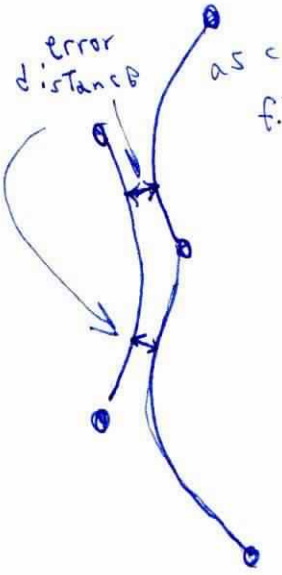




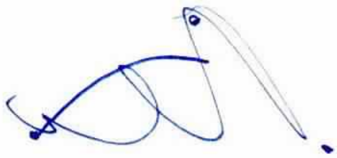
Intersection of
1&2
2&3 etc etc.

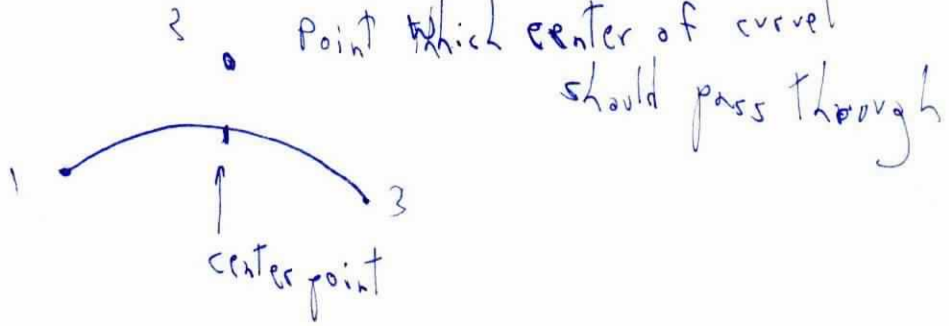




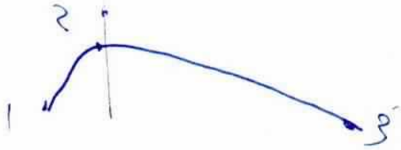


as compared with a fitting routine which passes through all points

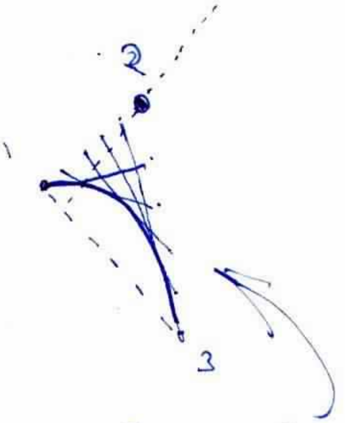




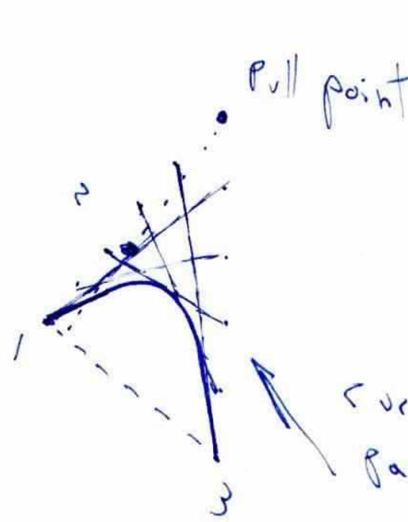
Point which center of curve should pass through



create an invisible pulling pt which will be placed on line passing through pt # 2 and is ⊥ To 1 & 2 until center of curve is close enough To it



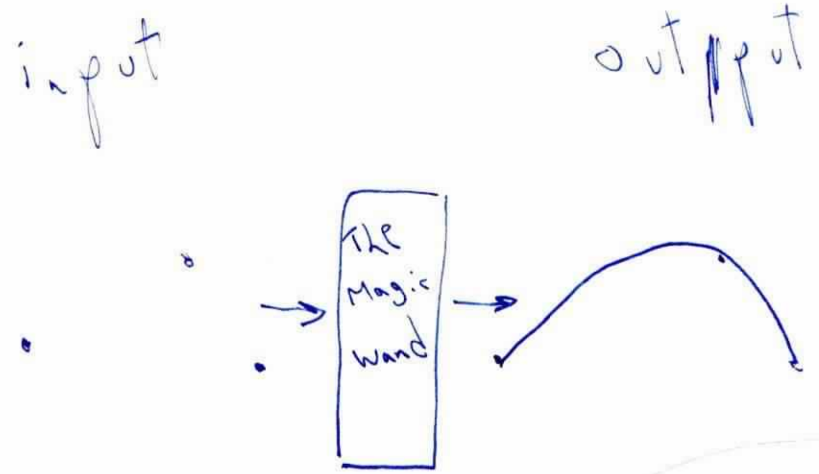
curve far from pt 2



for which actually curve is calculated.

curve closer to passing Through pt 2

pull point is to be generated automatically in fitting program. input would only be pts 1 2 & 3.



may be time consuming & clumsy

but is should work!

Painting

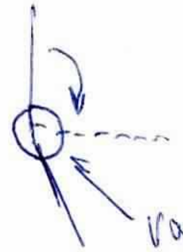
combine sections of canvass which have been plotted on with non plotted painting.

Document abstract expressionist type painting or write a program defining how to apply or throw paint.

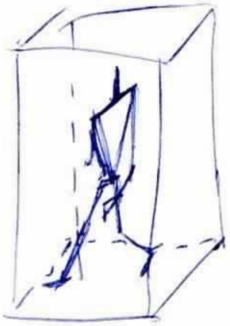
7/1

Dance

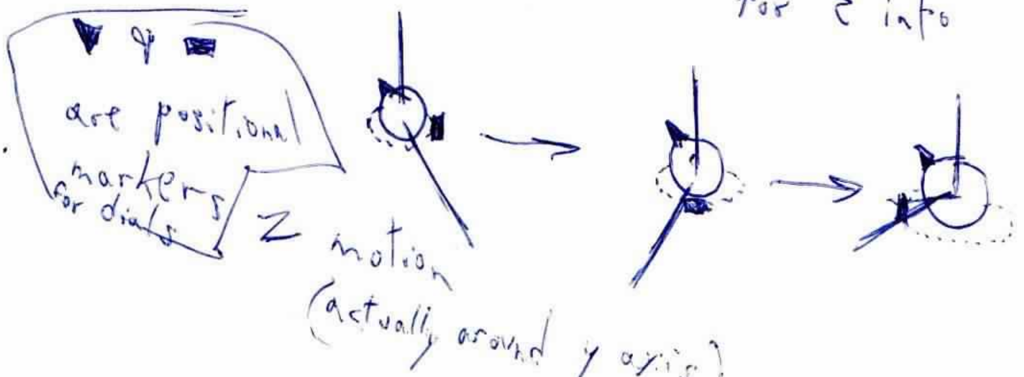
The manequin which would give positional information.



varied voltages ok for 2 dimensions but what about z



need 2 dials for each joint
1 for x y info 1 for z info



needs To be engineered out
as To how the connection between limbs
are To physically be made
Sonic pen kind of stuff.

Maybe build a manequin with protractors
or some kind of scale which could Then be
directly put into computer

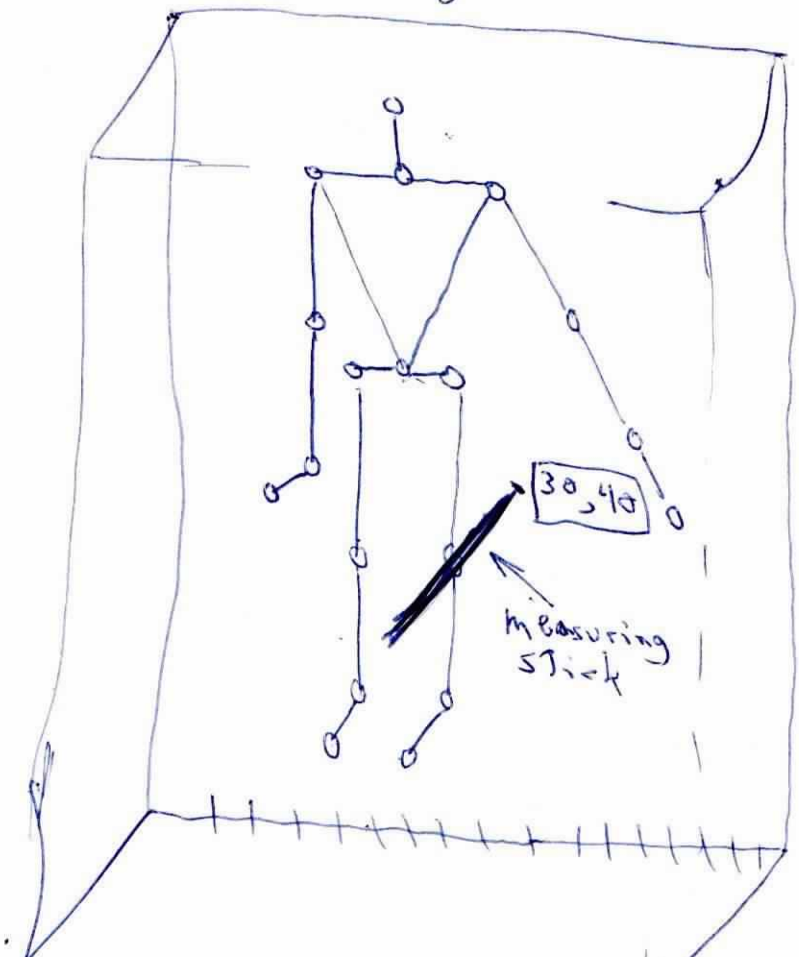
Like:

Command	Numerical Positioning
He	0
LS	10
RS	10

Kind of like drawing a picture by points but
a little better
You would position the dummy
get the numeric info then feed it in Quickly

and The position would be defined ~~with~~ with no putzing around with wrong angles and guessing ~~at~~ at angles and pos or neg signs.

Problem is to design an interfacable positional code readable from dummy and inputable to computer.



physical sticks with holes for Joints pass a stick through To read x y Then z position in box

---

may be clumsy in reading

x y z box

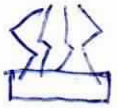
To Translate into computer usable data simple scale up or down ~~or~~ by some constant or not at all the xyz coordinate of box To The 1024, 780, 1024 of computer ⟶

Then in order To effect a position Just input the x y z data as point of the ~~b~~ variable names, ie. RVA consist of body points 4 & 5 with coordinates x y z, x'y'z'

Points 4 & 5 are The 1st & 2nd lines of the variable RVA and Just assign it The new values.

Scv's

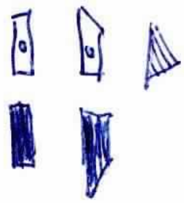
Make by constructing the random line out of ~~a~~ metal rod bent to shape. and set into some kind of base. ~~not~~ No connecting lines Therefore it would be unstable and vibrate about could be very large if use welded ~~[illegible]~~ rods



7/2 Dance

Exactly how do I make connection between Laban notation format and graphic, visual info of stick figure dancing.

First identify what body part(s) dealing with
Then determine what the movement is

Ranges of movement ie 

high, middle, and low major directional signs

each sign would mean something different for each body part

may involve some kind of path travel towards one of 27 points of reference (space cube)

or travel within 27 sections of space

These sections could be grouped

Top, middle, bottom layers ie deep middle & high of labanotation

Problem is that → is center of cube always good for location of joint? Doubt it and if not what happens to space cube.

Have to define movements of specific nature

The more specific the movement the easier but what about something like twist which has very complex results on different body parts

7/4

Home Video Syntdesizer manipulation

Switch to a channel / you have a bunch of knobs in front of you

would need a video camera

The synthesizer could be intefaced with a micro computer and users knobs go through The micro. Would be good for kids to play with instead of watching TV.

Dance -

· Need easy capability of displaying only part(s) of body like SAIL shit.

Make a display ~~global~~ system status which would control which part(s) is being displayes

More organization for the whole system is needed.

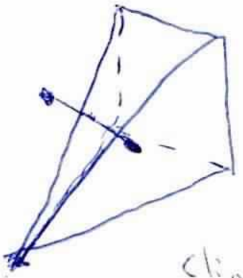
System Variables external to most programs

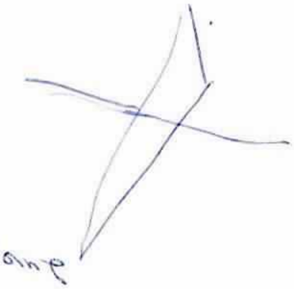
# of Interpolations

what to display

View point(s)

Clipping





clipping plane

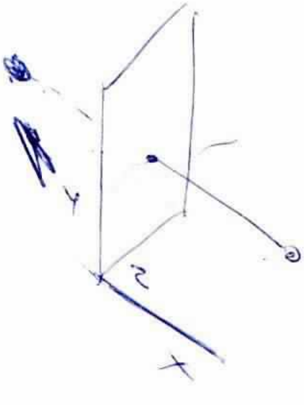
x = 0

$y = mx + b$

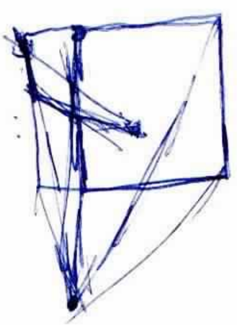
$mx = y - b$

$x = \frac{y-b}{m}$

Push an endpoint towards another



depending on which plane you intersect you know either an X or y value.

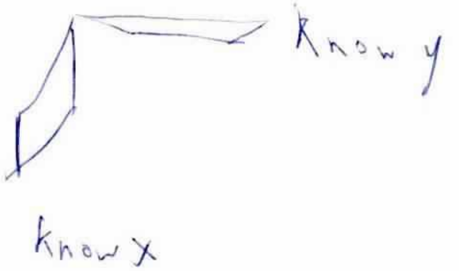


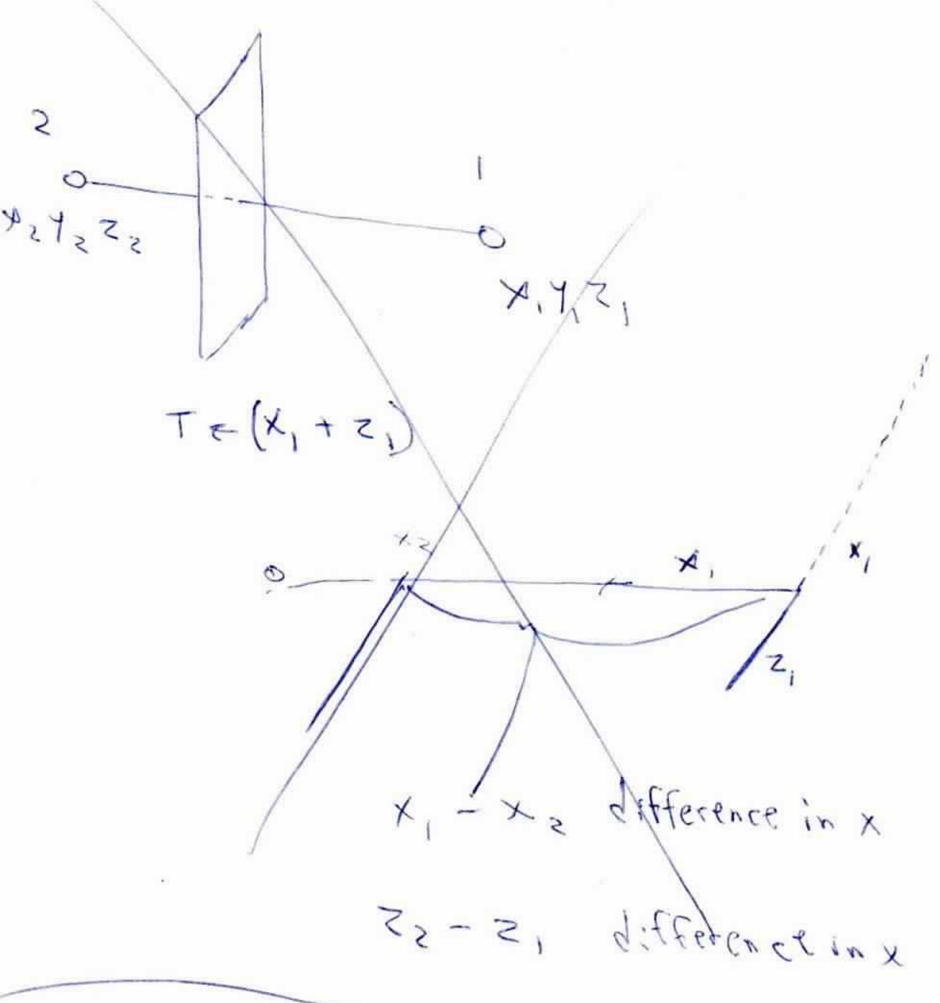
~~z you know~~

x you find
+
y " "

differen x plane

need x y z in clipping coordinate system 12-16 Pg 251



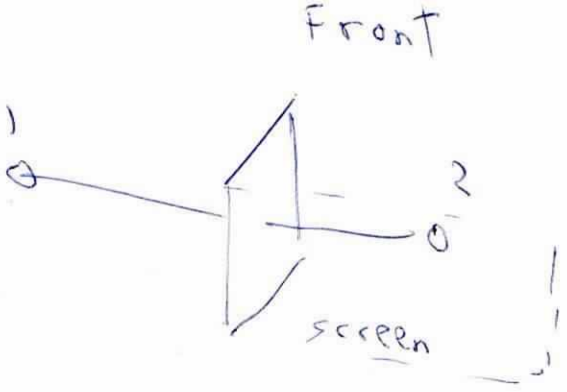


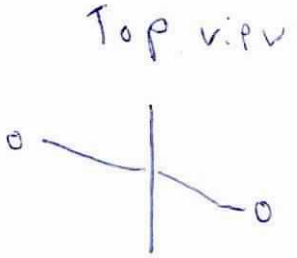
$T \leftarrow (x_1 + z_1)$

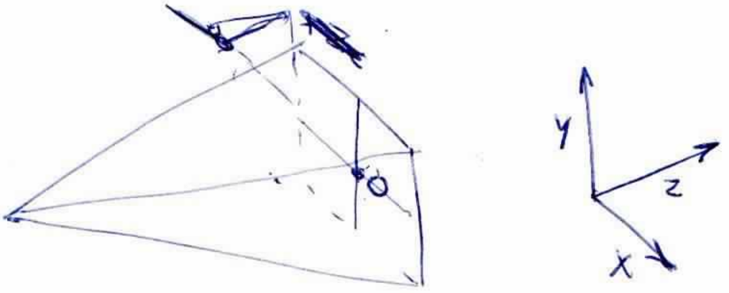
$x_1 - x_2$ difference in x

$z_2 - z_1$ difference in x

Point 1 must be point not on screen else switch 1 + 2


Front
1
2
screen


Top view


y
z
x

7/5 Dance

Create a random position ~~generation~~ generator!

Need to allow limbs to Travel anywhere within their sphere of motion and at first without constraints.

Constraints would be in ranges of angles of rot

Some constraints may be put in immediatly for very limited Joints. (Shoulders hip)

need a hieardy of rot.

As soon as any constraints put in then The motion of those limited joints is relative To other limbs of the body.

Automatic feeding of bodyparts to "Go" program and random X Y Z angles of rot

If range of randomness is incremented maybe some vaguely continuous movement would be possible.

Reset the DRL

but ~~inte~~ increment the range of angle chais

Can refine the random dance by controlling the amounts of randomness

ie The 10 1010 rand as opposed to The 100 100 100 rand

except of course in whatever angular term are worked out later.

Great potential for movies and mountains of positions.

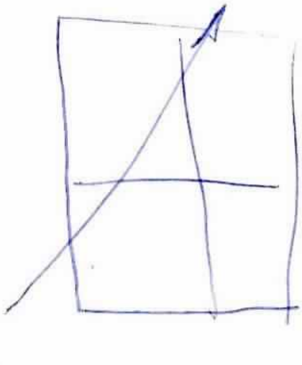
Random duets

Just make a hand(s) in contact and/or switch contact points (at random) clipping

get the a/b ratio from position of eye

eye[3] ÷ 390

?? where is the end z_c plane



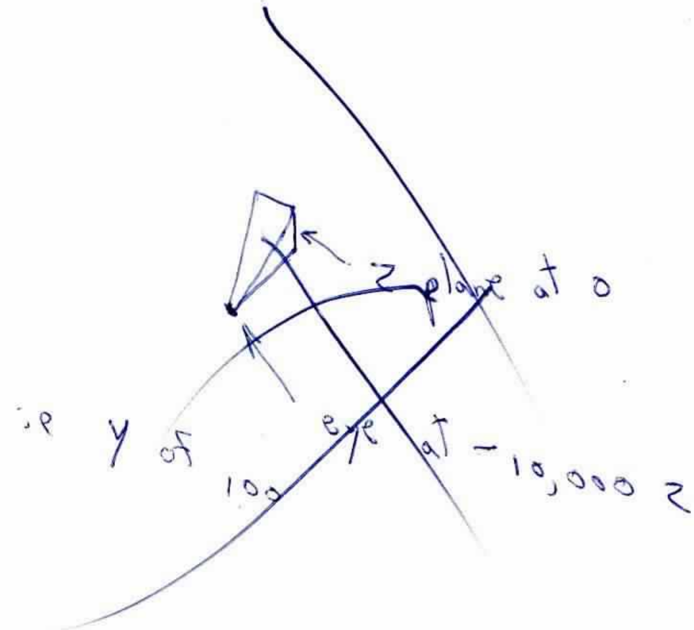
z_c & $-z_c$ seem to be acting as x values - & +

not taking those clipping limits from the z axis

~~its ok~~

~~$-z_c$ is minimum z and $+z_c$ is maximum z to be clipped against so fuck the signs~~

Tentative clipping routine after Sutherland-Cohen algorithm

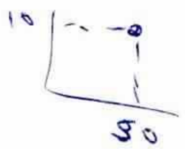


~~Mult~~ Plr

The plane z_c is characterized by a minimum and maximum ie Two "z" points ie a pos and neg value

Thus the $-z_c$ + $+z_c$

$-z_c = \overset{x}{0}, \overset{y}{0}$ $\quad +z_c = \overset{x}{1024}, \overset{y}{780}$



Take y of a point and compare it to the y of the z_c points $(1024, \overset{y}{780})$ 2 coordinates

Take y of a point and compare it To y of $-z_c$ $(0, \underset{y}{0})$

Take x of point compare to x of z_c $(\underset{x}{1024}, 780)$

Take x of point compare To x of $-z_c$ $(\underset{x}{0}, 0)$

Clipping after sutherland-Cohen algorithm

Z_c & $-Z_c$ are the ~~x~~ y clipping boundries

for APL variable will use

```
ZC & Z̲C

ZC ← 1024, 780
Z̲C ← 0, 0
```

1001	1000	1010
0001	0000	0010
0101	0100	0110

Compute Codes

```
C1 ← First point of line
C2 ← 2nd point of line     x & y are the pix[A; 2 3] respectively
              x ← C1[1] then x ← C2[1]
              y ← C1[2] Later y ← C2[2]     A is intremented To scan all points
→(x<Z̲C, x>ZC, y<Z̲C, y>ZC)/Left, Right, Bottom, Top

Left: code ← 1
→
Right: code ← 10
→
Bottom: code ← 100
→
Top: code ← 1000

~~C2 coder~~ → (ρcounter=1)/~~counter~~ c1coder
~~C2 code~~  → c2coder
             c1coder: c1 ← code

             c2coder: c2 ← code
```

7/6 Clipping

Way to scan picture matrix to get lines.

when scanning down increment a counter by one however if the second point has a 0 op code increment by two.

```
                                              ZC ← 1024,780
                                              ΞC ← 0,0
        Limit ← 1↑ρ Pixmat

→ Doit       counter ← 1
        Opchecker ← [counter+1;1]Pixmat                     If opc = 0
        counter ← Counter +      (1+ 0≠Opchecker)           get 2
        → (counter ≥ Limit)/stopscan
  Doit: Line ← [counter; 2 3 4],
                                  [counter+1; 2 3 4]
  ccounter ← 1
 just x&y →
        C1 ← Line[1 2]
        C2 ← Line[4 5]
  cchooser → ((ccounter=1),(ccounter=2))/c1coder, c2coder
  c1coder : x ← c1[1]      → coder
            y ← c1[2]        c2coder  x ← c2[1]
                                      y ← c2[2]
 coder: → ((x < ΞC[1]),(x > ZC[1]),(y < ΞC[2]),(y > ZC[2]))/Left, Right, Bottom, Top
  Left: code ← 1
  Right: code ← 10
  Bottom: code ← 100
  Top : code ← 1000   → ((ccounter=1),(ccounter=2))/ ~~c1←0, c2←0~~
    C1code ← code       → 2≠(ccounter ← ccounter+1)/c2coder
    →                   → (ccounter=1)/c2coder
```

```
Doit: Line ← [counter; 2 3 4], [counter+1; 2 3 4]
        ccounter ← 1
         c1 ← Line[1 2]
         c2 ← Line[4 5]

                                                    may be unnecessary Line
cchooser: → ((ccounter = 1), (ccounter = 2))/c1coder, c2coder

c1coder  x ← c1[1]
         y ← c1[2]
       → codes
c2 coder  x ← c2[1]
          y ← c2[2]

Coder:  code ← ((1×2X<≥c[1]) + (10×2X>2c[1]))
           code ← ((100×2Y<≥c[2]) + (1000×2Y>2c[2])) + code
    ~~Left~~
    ~~Right~~
    ~~Bottom~~
    ~~Top~~

~~→ (ccounter ← ccounter+1)~~
    ~~cchooser~~
   → ((ccounter=1), (ccounter=2))/c1co, c2co

   c1co: c1 ← code
      ccounter ← ccounter+1
    ~~→ c1coder~~
        → c2coder
   c2co: c2 ← code
// c1 & c2 are now coded To → #//
```

~~code ← ⍳ x (x ≤ zc[1], 10 × (~~ c1

~~code... → (Left, Right, In)[2 + x x~~

→(x ≤ zc[1])/Left

→(x > zc[1]/Right

→

~~code ← ⍳ x~~ maybe no iota

code ← ((1 × ? x < zc[1]) + (10 × ? x > zc[1]))

~~code ← 1×(x < zc[1]) + (10 × x > zc[1])~~

code ← (100×?y < zc[2] + 1000×?y > zc[2]), code

7/7/77

→(0 = c1 ∧ c2)/Showline

→(1 = c1 ∧ c2)/killine

→(c1 ≠ 0)/xyzget ~~swap~~

Swap: Temp ← c1, Line[1 2]

c1 ← c2

c2 ← Temp[1]

x1 ← Line[4]

y1 ← Line[5]

z1 ← Line[6]

x2 ← Line[1]

y2 ← Line[2]

z2 ← Line[3]

→Compute

xyzget: x1 ← Line[1]

y1 ← Line[2]

z1 ← Line[3]

x2 ← Line[4]

y2 ← Line[5]

z2 ← Line[6]

compute:

```
compute:        wrong condition but close (what if   1010
                                                     or
                                                     1001 etc)
Left: →(c1[4]≠1)/Right
      T ← (z1+x1)÷((x1-x2)-(z2-z1))
      z1 ← (T×(z2-z1))+z1
      x1 ← -z1
      y1 ← (T×(y2+y1))+y1
      → Recode
                                 →(c1[3]≠1)/Bottom
Right: ~~T ← (z1-x1)/(~~ ~~→(c1≠0)/Bottom~~
      T ← (z1-x1)÷((x2-x1)-(z2-z1))
      z1 ← (T×(z2-z1))+z1
      x1 ← z1
      y1 ← (T×(y2-y1))+y1
      → Recode
Bottom: ~~→(c1≠100)/top~~ →(c1[2]≠1)/Top
      T ← (z1+y1)÷((y1-y2)-(z2-z1))
      z1 ← (T×(z2-z1))+z1
      x1 ← (T×(x2-x1))+x1
      y1 ← -z1
      → Recode

Top: →(c1[1]≠1)/recode
      T ← (z1+y1)÷((y2-y1)-(z2-z1))
      z1 ← (T×(z2-z1))+z1
      x1 ← (T×(x2-x1))+x1
      y1 ← z1
      c1 ← code x1 y1
```

Make code a separate function.

```
R ← code x y

~~R ← ((1×⍳X<Z<[1])+(10×⍳X>Z<[1]))~~
~~R ← ((100×⍳Y<Z<[2])+(1000×⍳Y>Z<~~

R ← ((1×⍳X<Z<[1]),(10×⍳X>Z<[1]))
R ← ((100×⍳Y<Z<[2]),(1000×⍳Y>Z<[2])),R
```

7/8 Dance

Interaction of 2 bodys

Two bodys are in contact with each other at, at least one point. Problem is That The contact point(s) will vary ie hand → hand
hand → hip
hand → ?

Need some kind of dynamic control of contact points. One body should influence the position of The other. That's the point, otherwise you would Just be creating 2 sets of positions which match up. This matching up by hand (cursors) would be very clumsy. For starters maybe say that point x of body 1 is put in contact with point x' of body 2. ie a calculated MOVE This would be easier to handle if the movement are in general handled in

Scene and Time Units Type of system

ie Body 1 go from pos 10 → 20 in Time 100-300

Body 2 go from pos 15 → 25 in Time 150-300

IF This is done the reference to contact points would go:

Contact Body1 & Body 2 and points 4 + 7 ~~[illegible]~~

for positions
10-20 +
15-25
in Time
50-150

User would be led through The
questions for necessary info

One body could ~~influence~~ Truly influence another
(ie hand pushing an arm)
when The technique of ~~[illegible]~~ positioning the body
by having a lead part (hand) say reaching and The forearm
upper Arm and torso lean over and follow.

This gets into ~~[illegible]~~ The old problems of ~~[illegible]~~ hierarchy of stretch shit
and what has priority over what in following the lead.

Relation of one body To another is ~~[illegible]~~ would be basically same ~~as~~ to That of objects (props) in some respects not when body is walking on a props, Then it becomes like ground. Speaking of ground awarness.

Ground

Set a ground level (ie 0)

and say That Glevel ← 0

```
→(⌊/~~AllBps~~ Body[;3] < Glevel)/Mover
→0
```

A scan of all Bps (see The Room program)

```
Mover: ~~Body ←~~ 0
       ~~[illegible]~~ Glevel - ⌊/~~AllBps~~ Body[;3]
       YDif ← ⌊/AllBps[;3] ~~- Glevel~~
       Body ← (0, YDif, 0) Move Body
```

check if lowest point is above or below and move accordingly. Need a switch to turn off so body can Jump

```
Ground Glevel; Glevel; Lowbp; YDif ; sw

      sw ← Gswitch
      →(1↑1⌽sw='f')/0
→ (Lowbp ← ⌊/Body[;3]≠Glevel)/Mover
→0

Mover: ~~Glevel - Low~~
      YDif ← Glevel-Lowbp

Body ← (0, Ydif, 0) Move Body
```

Gswitch is a global charater variable 'on' or 'off'

Glevel a global variable = ground level

Just say Gswitch ← 'off' and you can Jump or fly

~~To be put in Go pr~~

To be referred To in Go program as a subroutine.

7/11

In Labanotation (LN) There are 2 different types of movement ~~[illegible]~~

1 movement towards a specific point (usually gestures)

2. movement away from a previous point of support

could be Taken as the shortest path To That point

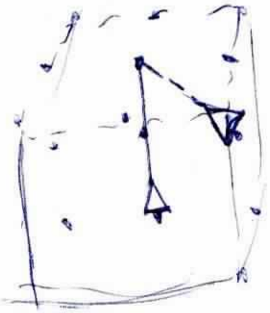
The path description itself. A more complex concept Then Just the movement towards a Point. Possibly The x y z angles could be a part of this.

A relative direction The ending depending only on rate + time allowed for motion,

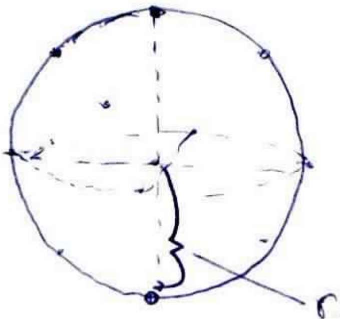
7/12

Do The 2 kinds of movement analysis require or desire 2 separate internal computer representations?

Destination analysis



length must be constant so the 27 points are not on a cube more like a sphere

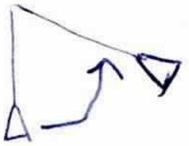


radius of reach or limb length

conceptually a cube with it's ~~A~~ implied directions is ok but for actual use those points are a direction and a distance (length of limb)

Input data is Destination + Time

Motion away from point



what is important here is the direction and how long and Fast The motion Takes

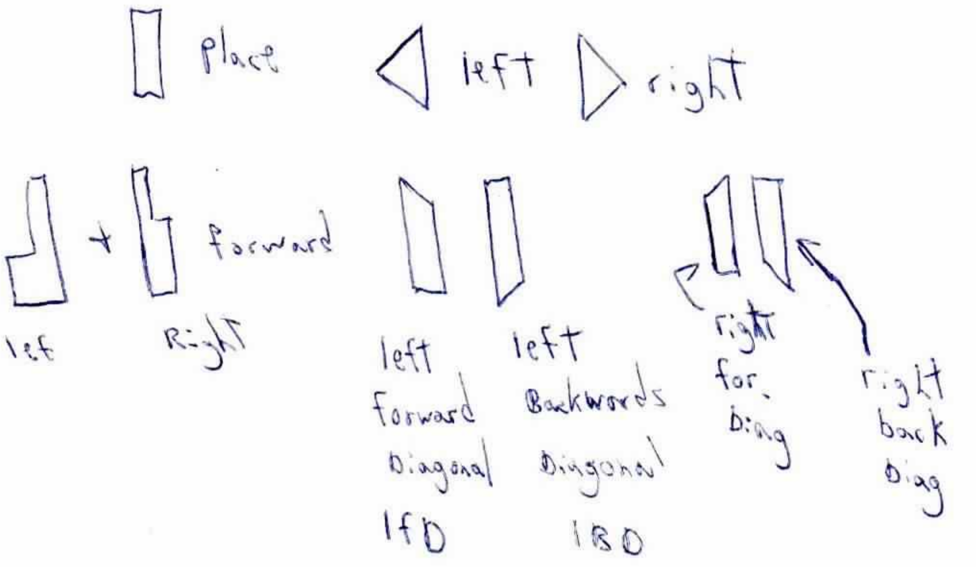
Input Data is

Direction Time + rate → $\frac{\text{distance}}{\text{Time}}$

Maybe two kinds of intefaces to the user would be good 2 sets of questions.

Labanotation symbols should be translated into x y z angles to do it and there will be a verbal interface to get positions. No x y z bullshit. Instead of 2 separate kinds of dialogue. One is ok which would give user choices as to what is to be input, computer will do the rest.

Must give names to Labanotation symbols for now just the major directions



all sighs
are

Up ▨ high or up

Mid ▯ middle

or

Down ▮ low or Down

If I say

Move $\frac{LA}{bp}$ $\frac{\text{Forward}}{\text{MaJor Direc}}$ Mid ← The left is known from the bp

The x y z angles could be computed
or maybe another method of movement
would be better

Place is Taken to be The Joint in This case the shoulder
Next we determine That Forward means the −z direction
and middle The same level as The Joint (shoulder level)
Arm is Then moved around x axis 90°

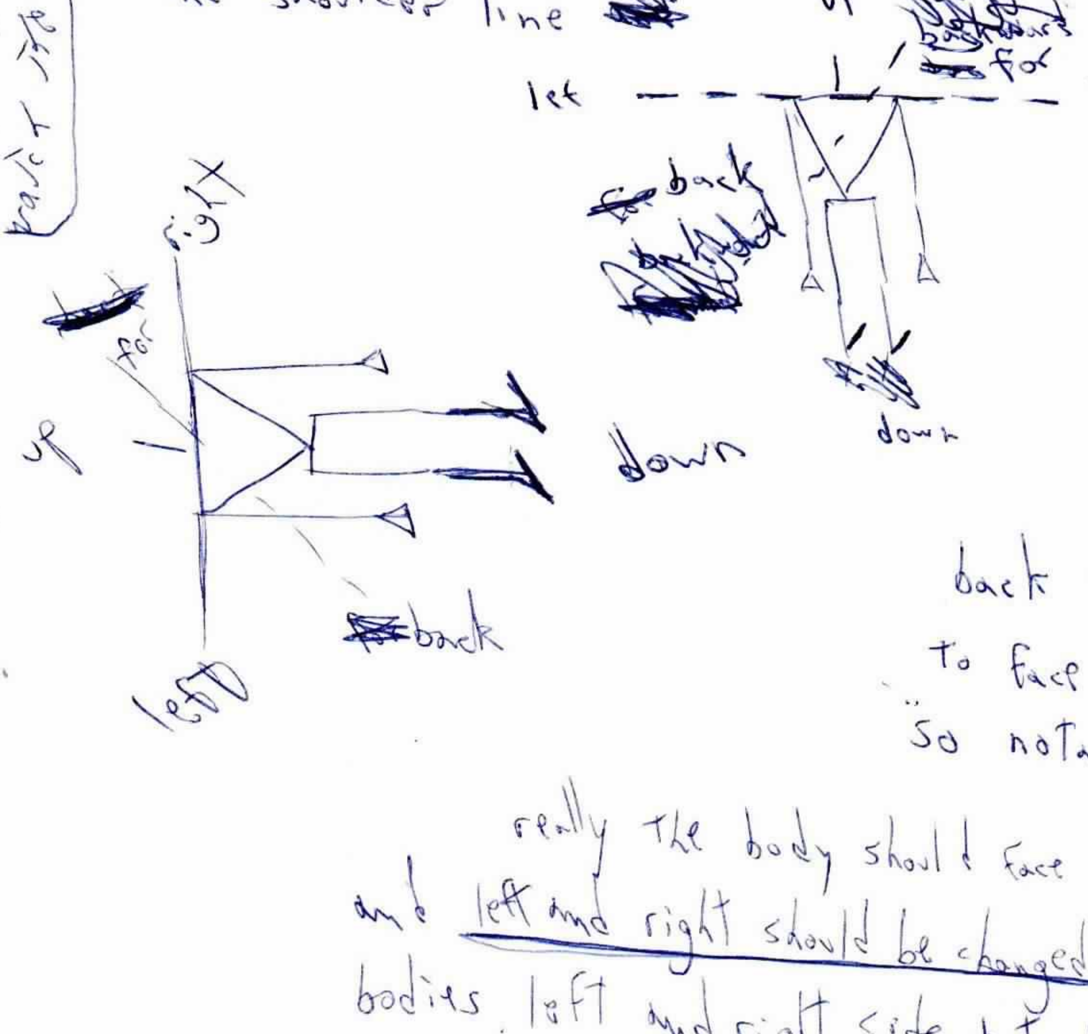
90 is determined by ~~difference~~ angle between
present pos and desired pos.

~~Translation table~~

~~(from LN to x y z angs~~

The Body axis is The reference and for now will be The shoulder. In Chap 25 Hutchinsen up and Down are always sky + ground but forward + backwards are at times the line of The head To ass. For now This is too complex and an absolute body axis will be assumed based on The shoulder line

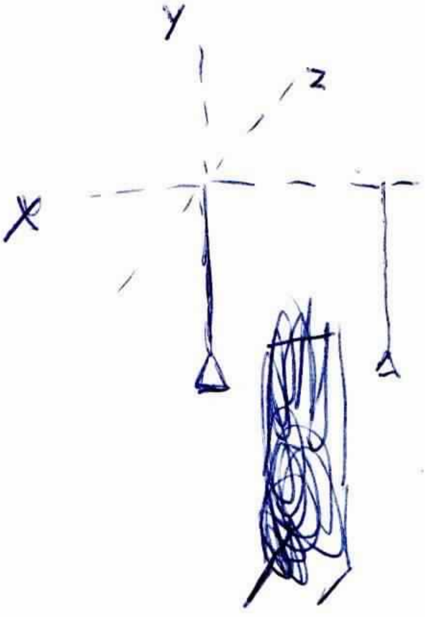
(point the feet)



back must be chaged
To face you (feet direction)
∴ So notation work might

really The body should face front and left and right should be changed to be the bodies left and right side not users

The APL bp names of left + right parts must be switched



For default position

left is +x

right is -x

forward is -z

back is +z

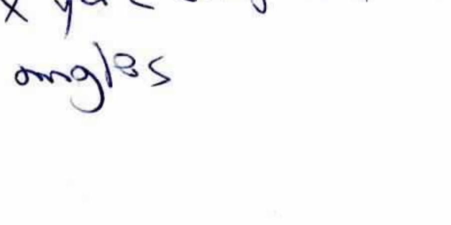
BAxis is based on shoulder line

if shoulder line ~~points~~ moves need

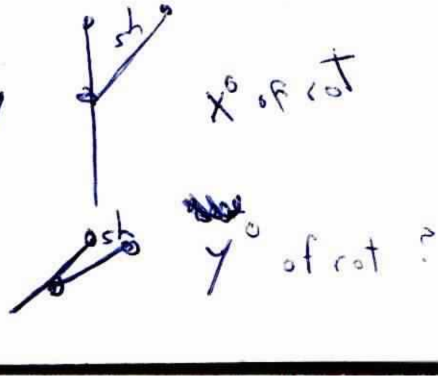
To know x y z angles of it movement

to change the ~~[illegible]~~ coordinate system

To find x y z change in shoulder line compare it To absolute axis.



x y a z component angles

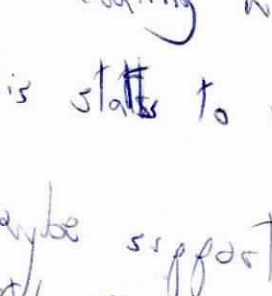


When shoulder line is moved
you find x y & z angles and then
when and bp is moved you first do This x y z rot
and then the movement. (This order may be incorrect or
not important)

Most of This is only good for gestures non support actions
when dealing with support forward means walk and
This starts to make things complicated.

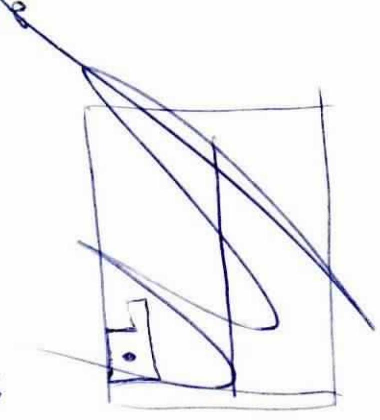
Maybe support actions such as walking could be define
with The gesture method.

Say That



means such and such
defined in terms
of gestures

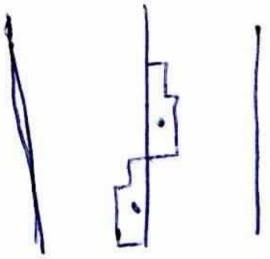
ie



Must first put in basics
the representation in words
ie
symbols
for each individual bp from There all other
movement sequences can be built up.

7/13 What exactly are these basics which would form a basic for any movement.?

First must see the difference between (LN) symbols in support and gesture columns.



A movement of support must end on ground

A gesture may or may not be on ground

support seems to be a sub group of all movements, which could be considered gestures of one kind or another

(LN) does not necessarily need a hierarchy of movement of body parts because of understood physiological limitations However I need this because of the lack of limitations. There is an implyed heirarchy of sorts in The set up of colums in The LN staff

Support movements can be treated with a destination method. But The destination is not stated and must be calculated.

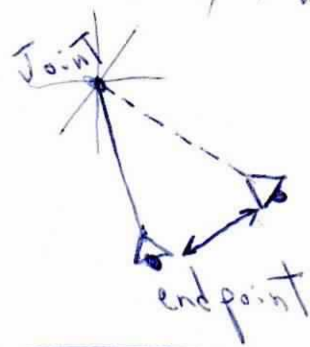
Lesson II Q's good?

→ supports are movements away from a starting point (motion)

The symbols in the support column mean a direction away from current position.

~~i.e. movement towards a point in space but not necessarily reaching~~ it.

what exactly is movement away from a point?



a direction and angle or distance or rate } all essentially same

Get symbols (word discussion)

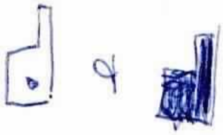
Determine whether a support or gesture

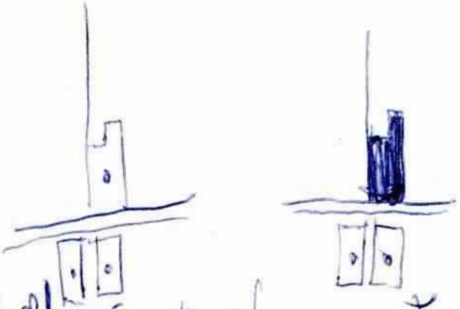
Differences in action

Direction is towards a point (from symbols) how much towards that point ~~from~~ but that point is always on ground and ~~might~~not be simply determined.

Distance is simply from where at to the point (gotten from symbol) and this has the ~~distance~~ direction implicitly in it

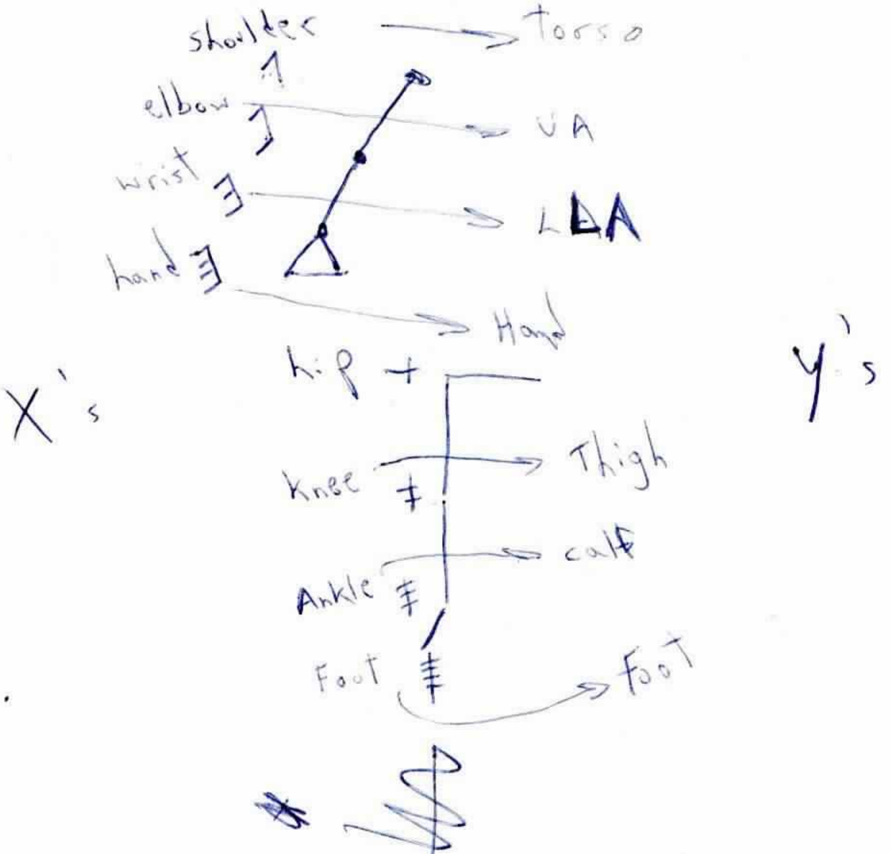
The points moved towards may be different in supports and gestures.

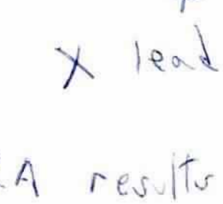
 for a step



both result in identical foot placement
it is the rest of body (center of gravity) which is different,
Different levels for support movements have a resultant complex' movement.

Back to basics.





X leads body part Y

ie Moving LLA results in symbol Ǝ

Each bp has an axis of space and sybolms ~~defininging~~ defining it going through its __fixed__ point ~~is~~ (The Joints as presently defined in computer)

center of axis is at first point of ~~bp~~ bp's .

is ~~a~~ -z a distance equal to length of bp

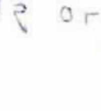
is +z

-x +x +y an equal length ~~[illegible] its around~~ -y

+x

~~[illegible]~~

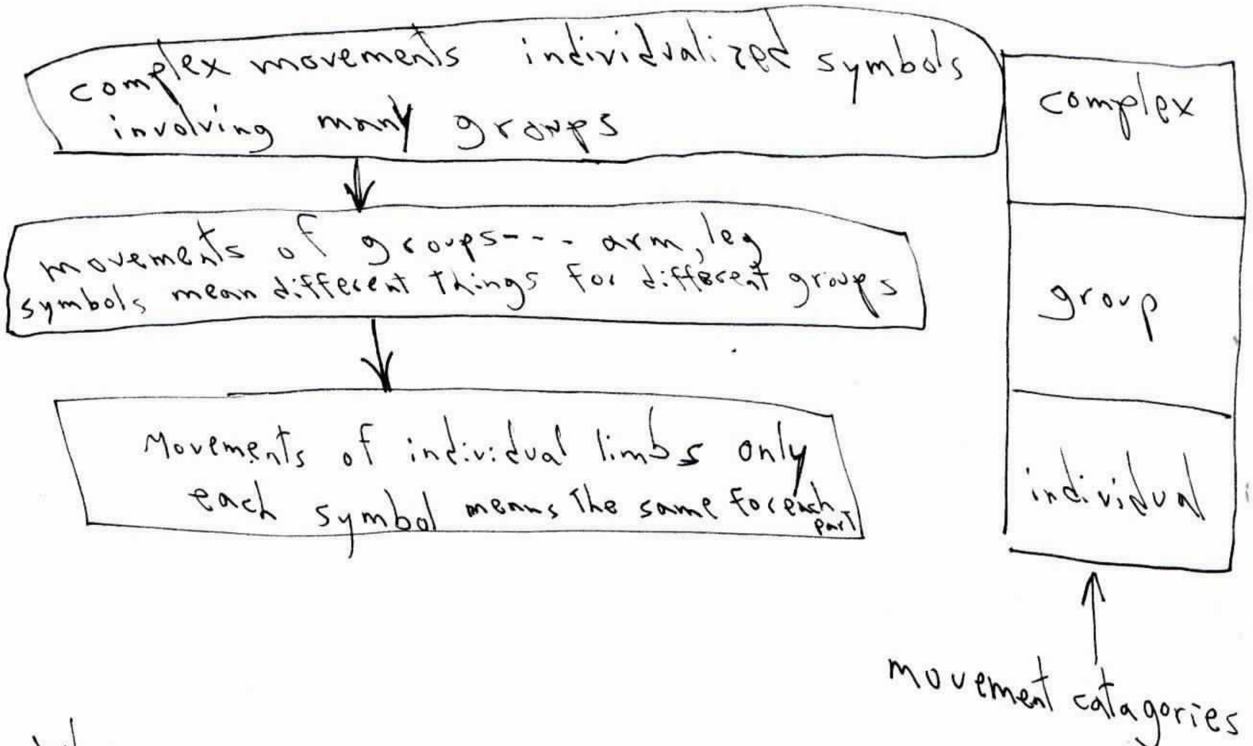
remember ~~the~~ stretch



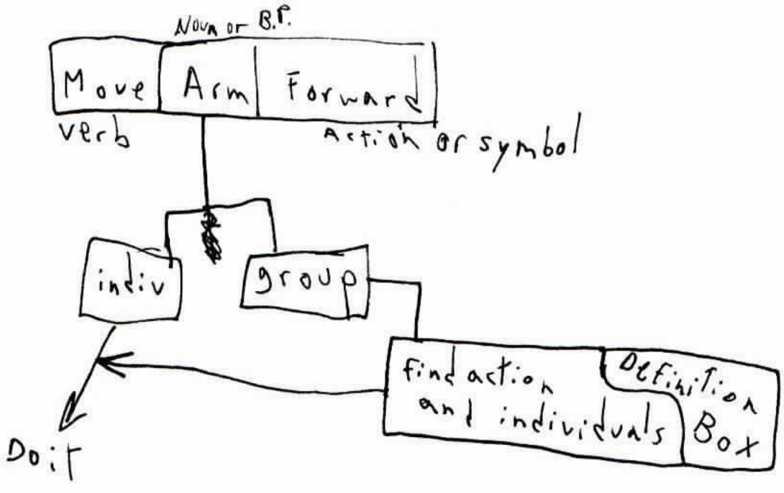
would stretch this much probably some constant like $\sqrt{2}$ or $\frac{\sqrt{3}}{2}$ ...

7/20 Dance

Need to organize the whole structure of the shtuff


complex movements individualized symbols involving many groups
movements of groups--- arm, leg
symbols mean different things for different groups
Movements of individual limbs only
each symbol means the same for each part
complex
group
individual
movement catagories

When you move a group you check a definition area which has in it The movement in terms of individual limbs.



This Definition Box
is The tough part.

In the definition box

Finding the individual bp's is no problem
doing it already

Defining the action is the problem.

supporting · &	gestures
~~[illegible]~~ directionaly oriented	movement towards a point
ie away from pt	
~~toward~~ ·	
in a direction	

~~Motion~~

~~[illegible]~~

7/24 Dance + Body Structure

Map or drape a 3D digitized model of human form this is an immobile sectional Treatment of body (Like Boeing Co.)

To make This body move with changes in structure (muscles) Map The moving points (control points) onto This figure and change The lines

This is a combination of my muscles algorithm with The static draping or mapping representation of The body.

Clothing and representation of it in computer.

Richer's Drawings show cloth as an outline, basically Trick is To find how This outline moves as body moves, The are connecting lines or points, ie waist. There would be limitations in The swing of cloth and it follows gravity sometimes, (when not directly being influenced by body) study The drawings

Dance

The conversational method of positioning the figure must make a distinction between supportive and gestural movement.

Maybe all leg. movement default to supportive until told otherwise — for starters

Time is necessary data for positioning a supportive movement

Need a conversational system to define basic movement which serve as the low level basics for complex movement ie walking

The basic definition system would allow quick easy definition of complex movements in terms of the elementary part which in tern are defined as x y z angles.

All basic definitions are treated as gestural for simplicities and practicality's sake
This should not affect the complex supportive movements

but if a supportive movement of a basic part ie foot or hand is used then there may be a conflict but to hell with it for now.

computer Painting 8/1

Program To tell how to paint

Divide canvas into #_____ sections

Use #___ Brush strokes in section 1 2 3... (a sequenc

stroke 1 is (color)

2

3

4

⋮

How big is canvas

□: 24 x 18

There will be #___ vertical lines

Line 1 will be ___" thick and ___" from left

Random Line dividing canvas - one color on one side and another on the other

---

~~Shade~~ Blend from one color to another

Pick a color Then refer to a list of possible colors to be ~~shaded~~ blended To

This avoids things like ~~shading~~ Blending from black to yellow or other screwed up ~~shadings~~ guess what

---

Place orthagonal lies in section #

Dance

Rod Rogers wants to use figure as a backround!

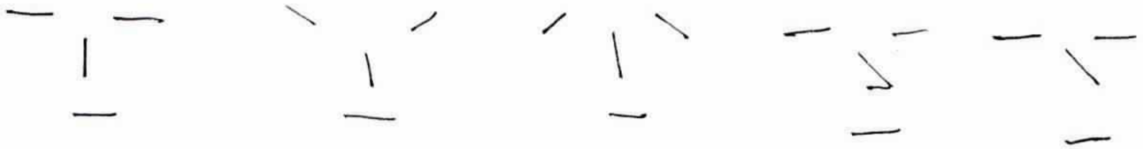
Make a film in a loop 1st pos the same as the last. So the time element isn't that much of a problem.

Must work on system for defining the basis of complex movements and more clearly defining what supportive movements are.
ie. So a walk up steps works well.

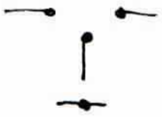
Problem is when there is a Transfer of support

Is it a simple linear shift of weight ie in walking

8/3 Computer Face



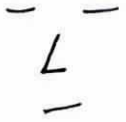
rot ~~tints~~ points are •



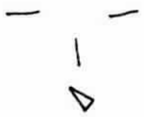
can make random Faces or expressions

may be usefull for psych studies

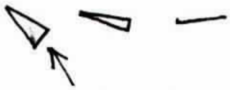
nose Trail



mouth trail

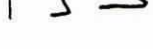


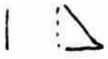
mouth



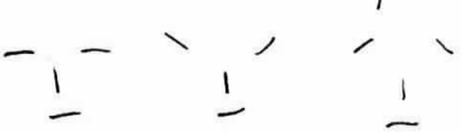
lenght of trails depends on deviation from horizontal

nose

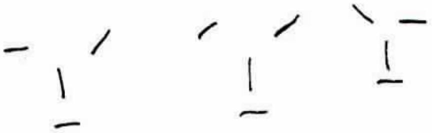


Trail goes to vertical drop from top 

can have a symetry switch (on or off) for eyes



switch on



switch off