ALTAMIRA (/)

ILITY ▶

(/blog/mobility)

Capture a Signature on iOS

11.06.12 │ **JASON HARWIG (/profiles/jharwig)** │ Employee Post (/blog/employee-posts)

Tweet ⟨8   Like ⟨0   **8+1** ⟨8

The Square Engineering Blog has a great article on Smoother Signatures (http://corner.squareup.com/2012/07/smoother-signatures.html) for Android, but I didn't find anything specifically about iOS. So, what is the best way to capture a users signature on an iOS device?

Although I didn't find any articles on signature capture, there are good implementations on the App Store. My target user experience was the iPad application Paper by 53 (http://fiftythree.com/paper), a drawing application with beautiful and responsive brushes.

All code is available in the Github repository: SignatureDemo (https://www.github.com/jharwig/SignatureDemo).

Connecting the Dots

Touch Points
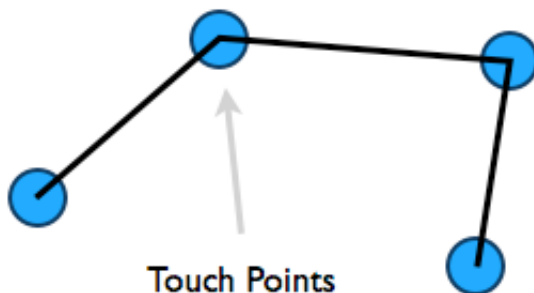
The simplest approach is to capture the touches and connect them with straight lines.

In the initializer of a `UIView` subclass, create the path and gesture recognizer to capture touch events.

```
// Create a path to connect lines
path = [UIBezierPath bezierPath];

// Capture touches
UIPanGestureRecognizer *pan = [[UIPanGestureRecognizer alloc] initWithTar
get:self action:@selector(pan:)];
pan.maximumNumberOfTouches = pan.minimumNumberOfTouches = 1;
[self addGestureRecognizer:pan];
```

Capture the pan events into a bézier path by connecting the points with lines.

```
- (void)pan:(UIPanGestureRecognizer *)pan {
    CGPoint currentPoint = [pan locationInView:self];

    if (pan.state == UIGestureRecognizerStateBegan) {
        [path moveToPoint:currentPoint];
    } else if (pan.state == UIGestureRecognizerStateChanged)
        [path addLineToPoint:currentPoint];

    [self setNeedsDisplay];
}
```

Stroke the path

```
- (void)drawRect:(CGRect)rect
{
    [[UIColor blackColor] setStroke];
    [path stroke];
}
```
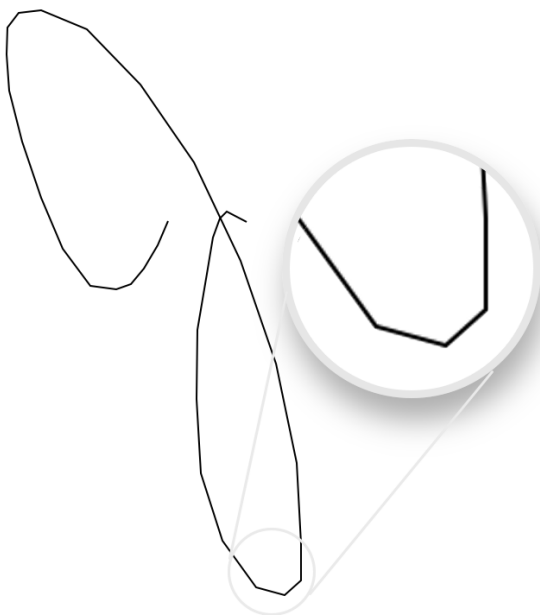

An example "J" character rendered using this technique reveals some issues. At slow velocities iOS captures enough touch resolution that the lines aren't noticeable, but faster movement shows large gaps between touches that accentuates the lines.

The 2012 Apple Developer Conference included a session Building Advanced Gesture Recognizers (https://developer.apple.com/videos/wwdc/2012/?id=233) that addresses this issue using math.

Quadratic Bézier Curves
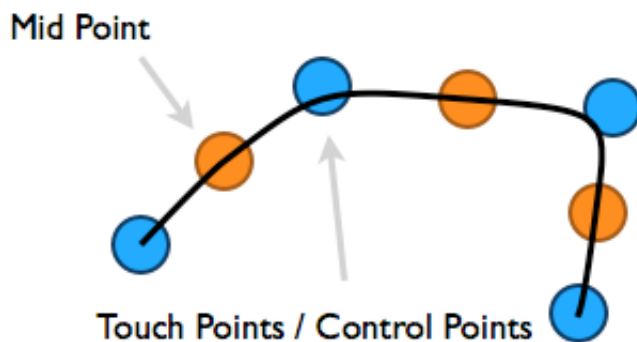
Instead of connecting the lines using touch points we'll use quadratic bézier curves to connect the points using the technique discussed in the aforementioned WWDC session (Seek to 42:15.) Connect the touch points with a quadratic curve using the touch points as the control points and the mid points as start and end.

Adding quadratic curves to the previous code requires the storing the previous touch point, so add an instance variable for that.

```
CGPoint previousPoint;
```

Create a function to calculate the midpoint of two points.

```
static CGPoint midpoint(CGPoint p0, CGPoint p1) {
    return (CGPoint) {
        (p0.x + p1.x) / 2.0,
        (p0.y + p1.y) / 2.0
    };
}
```

Update the pan gesture handler to add quadratic curves instead of straight lines

```
- (void)pan:(UIPanGestureRecognizer *)pan {
    CGPoint currentPoint = [pan locationInView:self];
    CGPoint midPoint = midpoint(previousPoint, currentPoint);

    if (pan.state == UIGestureRecognizerStateBegan) {
        [path moveToPoint:currentPoint];
    } else if (pan.state == UIGestureRecognizerStateChanged) {
        [path addQuadCurveToPoint:midPoint controlPoint:previousPoint];
    }

    previousPoint = currentPoint;

    [self setNeedsDisplay];
}
```
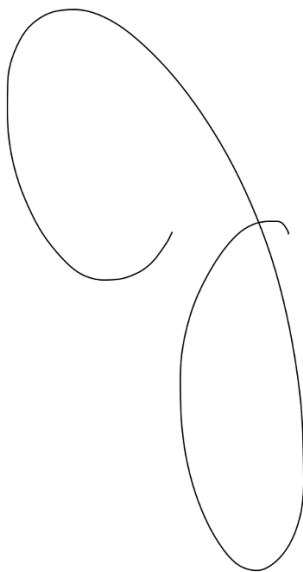
Not much code and already we see a big difference. The touch points are no longer visible, but it looks a little bland when drawing a signature. Every curve is the same width, which doesn't match the physics of a real pen.

Variable Stroke Width

The width can be varied based on the touch velocity to create a more natural stroke. The `UIPanGestureRecognizer` already includes a method called `velocityInView:` that returns the current touch velocity as a `CGPoint`.

To render a stroke of varying width, I switched to OpenGL ES and a technique called tesselation to convert the stroke into triangles – specifically, triangle strips (OpenGL has support for drawing lines, but iOS doesn't support variable line widths with smoothing.) The quadratic points along a curve also need to be calculated, but is beyond the scope of this article. Check the source on github (https://www.github.com/jharwig/SignatureDemo) for details.

Given two points, a perpendicular vector is calculated and its magnitude set to half the current thickness. Given the nature of `GL_TRIANGLE_STRIP` only two points are needed to create the next rectangle segment with two triangles.

Quadratic Bézier Points   Triangle Strip Points
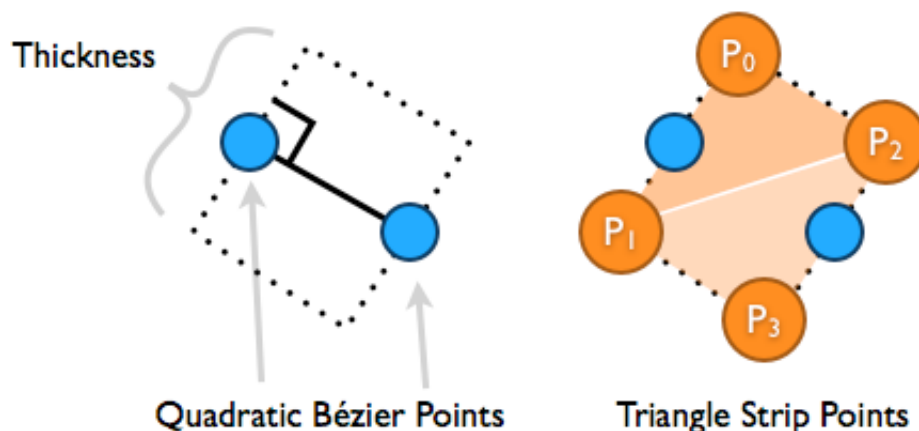
Here is an example of the final output using quadratic bézier curves, and velocity based stroke thickness creating a visually appealing and natural signature.
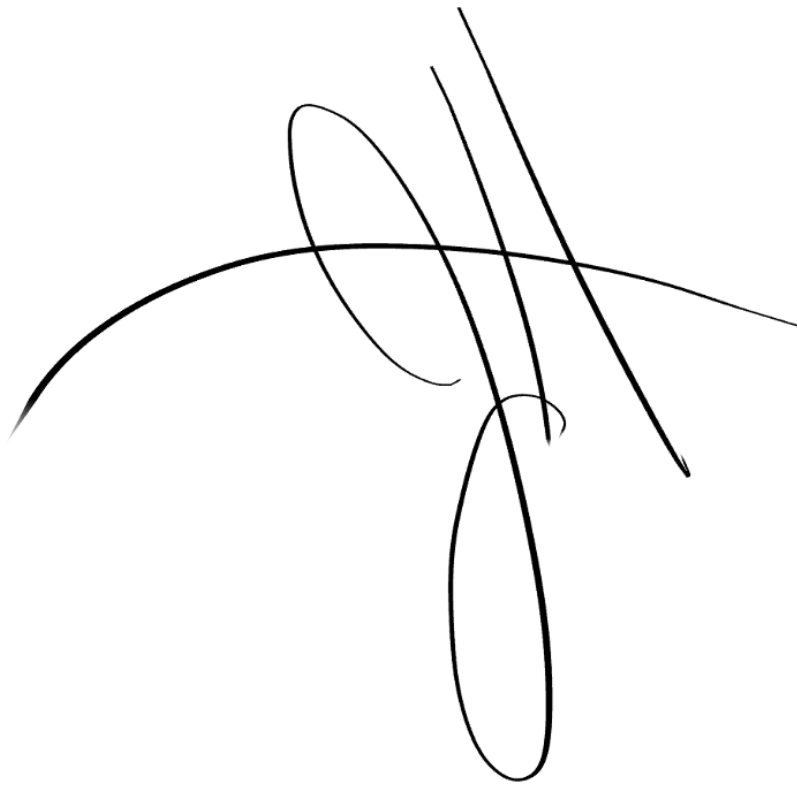
Updated on: 06.13.2013

TAGS: IOS (/blog/tag/ios)   OPENGL (/blog/tag/opengl)

COMMENTS

Smaquois said:

Very nice. I've seen a few other tutorials along the same lines but none as thourough as yours.

11.10.12   Reply

> **Thordin** said:
>
> Along the same "lines" heheh
>
> 03.22.13   Reply

Adam Wulf said:

this post is great, thanks! I've had trouble finding such a simple explanation for variable width in OpenGL. For your code - the License states the copyright must be included in all copies of the software - i assume that means all copies/modifications of the src? Any notification needed if this is included in an app store binary build?thanks again!

12.31.12   Reply

jharwig said:

Glad it's useful, the license doesn't require attribution. But, it's appreciated if you'd like to.

01.02.13   Reply

Adam Wulf said:

Another thought - could this be adapted to use variable alpha instead of variable width? Thanks!

12.31.12   Reply

jharwig said:

Yes, you'd just need a data structure of alpha values that you apply to the triangles. I have some other requests, so maybe I'll do an update with some of these modifications.

01.02.13   Reply

kel said:

This is awesome. Can I add your signature view anywhere I'd like in an application?

01.11.13   Reply

Kyle Clegg said:

Sorry being a n00b, but I can't figure out how to use it, even when using your NISignatureView directly from github. I tried adding a UIView and setting its class to NISignatureView, matching your sample project, and nothing happens. I tried hooking up the delegate but also saw no change. Could you add a few more details or update the README on github to show how to implement it from ground zero?

01.19.13   Reply

Kjell said:

Great post, thanks! I noticed that a dot isn't drawn. This dot should have the same width as the last drawn line, I think. Maybe for a future update? :)

03.05.13   Reply

jharwig said:

I've added support for dots in the github repository

04.10.13   Reply

James Eberhardt said:

Pretty cool. I've added a method to return a PNG of the signature. Not sure how to add it to GitHub, but if that's possible, please do so, cause I haven't added anything to Git before. -(UIImage*) imageRepresentation {

```
  UIGraphicsBeginImageContext(self.bounds.size);

  [self.layer renderInContext:UIGraphicsGetCurrentContext()];

  UIImage *image = UIGraphicsGetImageFromCurrentImageContext();

  UIGraphicsEndImageContext();

  return image;

 }
```

04.10.13   Reply

Smin Rana said:
Very nice!

04.19.13   Reply

Janakiraman Rajendran said:
Great !!!

04.22.13   Reply

김응식(EungShik Kim) said:
Wow! It's just great!

04.22.13   Reply

Chris said:
this is the best tutorial I have seen on this subject, brilliant

04.22.13   Reply

Chris said:
Thanks for such a clear and excellent tutorial!

Any thoughts on how a texture could be implemented into this? (ex: pencil or brush stroke instead of a
pen)

Thanks again!

07.09.13   Reply

Denis said:
Excellent! Thank you!

07.31.13   Reply

Brian Jiang said:

Thanks a ton!

08.04.13   Reply

ngs_mtech said:

Thank you, I'd six month searching for this pretty clean solution.

Regards.

09.06.13   Reply

Dmitry Nadezhdin said:

Finally someone explains it properly! Thanks a lot for taking your time to post this, it seems like this will stay relevant for quite a while.

09.26.13   Reply

Akash said:

Hi, Thanks for the great tutorial. I just love tutorial.

I am stuck at one problem. How do I make background of sign transparent.

Thanks.

09.30.13   Reply

Aldo Simoncini said:

how can I load an image and set it as the background of GLKView? thanks! :)

10.08.13   Reply

New Comment

Author

Body

Markdown Supported

Comment   Toggle Preview

ANALYTICS (/blog/analytics)

VISUALIZATION (/blog/visualization)

COMMS & SENSORS (/blog/comms-sensors)

CYBER (/blog/cyber)

MOBILITY (/blog/mobility)

BIG DATA (/blog/big-data)

Join Us

All Source/Targeting Analyst (https://www.altamiracorp.com/job/1442)

Washington, DC

Responsibilities: These analysts possess the capability of fusing intelligence ...
(https://www.altamiracorp.com/job/1442)

Computer Network Operations (CNO) Analyst/Engineer I (https://www.altamiracorp.com/job/1167)

Annapolis Junction, MD

Altamira is seeking a Computer Network Operations (CNO) Analyst to support our ...
(https://www.altamiracorp.com/job/1167)

View All (/jobs)

More By Jason

Mock NSDate (or other) class methods (/blog/employee-posts/mock-nsdate-or-other-class-met)
Jason Harwig (https://www.altamiracorp.com/profiles/jharwig) on 02.29.12

HTML5 and CSS3 talk at RubyNation (/blog/employee-posts/html5-and-css3-talk-at-rubynat-1)
Jason Harwig (https://www.altamiracorp.com/profiles/jharwig) on 04.02.11

"The Daily" Image Carousel using Core Animation (/blog/employee-posts/coreanimation-image-

carousel)
Jason Harwig (https://www.altamiracorp.com/profiles/jharwig) on 02.08.11

Related by Category

Android Hacking 101 (/blog/employee-posts/android-hacking-101)
Stephen Mouring Jr. (https://www.altamiracorp.com/profiles/smouring) on 02.09.13

Creating and Using Static Libraries for iPhone using Xcode 4.3 (/blog/employee-posts/creating-and-using-static-libraries-for-iphone-using-xcode-4)
Tyler Vernon (https://www.altamiracorp.com/profiles/tvernon) on 07.17.12

Activities vs. Fragments in Android (/blog/employee-posts/fragments-vs-activities-in-and)
Tom Neumark (https://www.altamiracorp.com/profiles/tneumark) on 05.31.12

CAPABILITIES

ANALYTICS (/what-we-do/analytics)

VISUALIZATION (/what-we-do/visualization)

COMMS & SENSORS (/what-we-do/comms-sensors)

CYBER (/what-we-do/cyber)

MOBILITY (/what-we-do/mobility)

BIG DATA (/what-we-do/big-data)

RESOURCES

Employee Posts (/blog/employee-posts)

News (/blog/news)

Published Works (/blog/published-works)

Speaking Engagements (/blog/speaking-engagements)

Tech Talks (/blog/tech-talks)

Case Studies (/blog/case-studies)

CONNECT

Twitter (https://twitter.com/explorealtamira)

Facebook (https://www.facebook.com/AltamiraTechnologiesCorporation)

YouTube (https://www.youtube.com/user/nearinfinity)

LinkedIn (https://www.linkedin.com/company/altamira-corporation)

CONTACT

(/contact-us)

8201 Greensboro Drive
Suite 800
McLean, Va 22102

📞 703.813.2100

🖨 703.813.1740

✉ info@altamiracorp.com (mailto:info@altamiracorp.com)

amazon
web services | Partner
Network

CONSULTING PARTNER