

给绘图添加过程动画

1. 简述：在使用 CoreGraphics 绘制图形完成后，有时候我们希望绘制完成的图形能够动起来，或者希望能够是动画式的绘制。

2. 实现：

在 drawrect 方法里实现的 CoreGraphics 绘图暂时没找出合适的办法实现这种效果。以下介绍下另一种 CoreAnimation 实现的绘制，那就是 CAShapeLayer 结合 CGContextPath 实现的绘图。以下为示例代码：

```
#import "ViewController.h"

@interface ViewController ()

@property (strong, nonatomic) CAShapeLayer *shapLayer;

@end

@implementation ViewController

- (void)viewDidLoad {
    [super viewDidLoad];
    self.shapLayer = [CAShapeLayer layer];
    self.shapLayer.frame = CGRectMake(0, 300, CGRectGetWidth([UIScreen mainScreen].bounds), 40);

    self.shapLayer.lineWidth = 1;
    self.shapLayer.strokeColor = [UIColor redColor].CGColor;
    self.shapLayer.fillColor = [UIColor clearColor].CGColor;

    [self.view.layer addSublayer:self.shapLayer];

    [self drawPath];
    // Do any additional setup after loading the view, typically from a nib.
}

- (void)viewDidAppear:(BOOL)animated {
    [super viewDidAppear:animated];
    [self startAnimation];
}

- (void)touchesBegan:(NSSet<UITouch *> *)touches withEvent:(UIEvent *)event {
    [self startAnimation];
}
```

只需将绘好的 path 赋给 shapLayer 的 path，图形就绘制到 shapLayer 上了。具体绘制可以采用 UIBezierPath 或 CGMutablePathRef 等。如下所示：

```
- (void)drawPath{
    //用贝塞尔曲线绘制
    UIBezierPath *path = [UIBezierPath bezierPath];
    [path moveToPoint:CGPointMake(0, 40 * sin( 2.0 * M_PI / 60 * 0))];
    for (int i = 1; i < CGRectGetWidth([UIScreen mainScreen].bounds); i++) {
        CGFloat y = 40 * sin( 2.0 * M_PI / 60 * i );
        [path addLineToPoint:CGPointMake(i, y)];
    }

    self.shapLayer.path = path.CGPath;

    //使用CoreGraphic库方法绘图
    //  CGMutablePathRef path1 = CGPathCreateMutable();
    //  CGPathMoveToPoint(path1, nil, 0, 40 * cos( 2.0 * M_PI / 60 * 0));
    //  for (int i = 1; i < CGRectGetWidth([UIScreen mainScreen].bounds); i++) {
    //      CGFloat y = 40 * cos( 2.0 * M_PI / 60 * i );
    //      CGPathAddLineToPoint(path1, nil, i, y);
    //  }
    //
    //  self.shapLayer.path = path1;
}
```

这样绘制出来的是静态的，那么如何让它动起来呢？很简单，使用 CoreAnimation 给 shapeLayer 动画即可。下面就实现绘制过程动画为例，实现如下：

```
- (void)startAnimation{
    CABasicAnimation *animation = [CABasicAnimation animationWithKeyPath:@"strokeEnd"];
    animation.duration = 2.5f;
    animation.fillMode = kCAFillModeForwards;
    animation.removedOnCompletion = NO;
    animation.fromValue = @0;
    animation.toValue = @1;
    animation.timingFunction = [CAMediaTimingFunction functionWithName:kCAMediaTimingFunctionEaseInEaseOut];
    [self.shapLayer addAnimation:animation forKey:@"dfas"];
}
```

具体效果可以参照功能演示。