多屏幕适配
根据不同屏幕的尺寸比例进行等比例缩放处理，即重写CGRectMake方法，即CGRectMakeAutolayout。

示例
方法1
```
UIImageView *imageViewTmp = [[UIImageView alloc]
initWithFrame:CGRectMakeAutolayout(10.0, 10.0, 30.0,
30.0, YES, YES)];
 [self.view addSubview:imageViewTmp];
imageViewTmp.backgroundColor = [UIColor
colorWithWhite:0.0 alpha:0.2];
imageViewTmp.contentMode =
UIViewContentModeScaleAspectFit;
```

方法2
```
#define sizeImage GetHeightAutolayout(40.0)
UIImageView *imageView = [[UIImageView alloc]
initWithFrame:CGRectMake(10.0, 10.0, sizeImage,
sizeImage)];
 [self.view addSubview:imageView];
imageView.backgroundColor = [UIColor
colorWithWhite:0.0 alpha:0.2];
imageView.contentMode =
UIViewContentModeScaleAspectFit;
```

实现方法
```
////////////////////////////////////////////////////
/////////////////////////////

/*
 各版本尺寸
 1  iPhone4      640*960   320*480 2倍
```

2 iPhone4S   640*960   320*480 2倍
3 iPhone5   640*1136 320*568 2倍
4 iPhone5S   640*1136 320*568 2倍
5 iPhone5C   640*1136 320*568 2倍
6 iPhone6   750*1334 375*667 2倍
7 iPhone6 Plus 1242*2208 414*736 3倍

各版本比例
iPhone5,   AutolayoutScaleX = 1, AutolayoutScaleY = 1;
iPhone6,   AutolayoutScaleX = 1.171875,
AutolayoutScaleY = 1.17429577;
iPhone6Plus, AutolayoutScaleX = 1.29375,
AutolayoutScaleY = 1.295774;
*/

```
#define IS_IPAD_Autolayout
(UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad)
#define IS_IPHONE_Autolayout
(UI_USER_INTERFACE_IDIOM() ==
UIUserInterfaceIdiomPhone)
#define IS_RETINA_Autolayout        ([[UIScreen
mainScreen] scale] >= 2.0)

#define SCREEN_WIDTH_Autolayout      ([[UIScreen
mainScreen] bounds].size.width)
#define SCREEN_HEIGHT_Autolayout     ([[UIScreen
mainScreen] bounds].size.height)
#define SCREEN_MAX_LENGTH_Autolayout
(MAX(SCREEN_WIDTH_Autolayout,
SCREEN_HEIGHT_Autolayout))
#define SCREEN_MIN_LENGTH_Autolayout
(MIN(SCREEN_WIDTH_Autolayout,
SCREEN_HEIGHT_Autolayout))
```

```objc
#define IS_IPHONE_4_OR_LESS_Autolayout
(IS_IPHONE_Autolayout && SCREEN_MAX_LENGTH_Autolayout
< 568.0)
#define IS_IPHONE_5_Autolayout
(IS_IPHONE_Autolayout && SCREEN_MAX_LENGTH_Autolayout
== 568.0)
#define IS_IPHONE_6_Autolayout
(IS_IPHONE_Autolayout && SCREEN_MAX_LENGTH_Autolayout
== 667.0)
#define IS_IPHONE_6P_Autolayout
(IS_IPHONE_Autolayout && SCREEN_MAX_LENGTH_Autolayout
== 736.0)

////////////////////////////////////////////////
/////////////////////////////

#define Delegate_Autolayout    ([[UIApplication
sharedApplication] delegate])
#define ScreenWidth_Autolayout ([[UIScreen mainScreen]
bounds].size.width)
#define ScreenHeight_Autolayout ([[UIScreen mainScreen]
bounds].size.height)

#define ScaleX_Autolayout ((ScreenHeight_Autolayout >
480.0) ? (ScreenWidth_Autolayout / 320.0) : 1.0)
#define ScaleY_Autolayout ((ScreenHeight_Autolayout >
480.0) ? (ScreenHeight_Autolayout / 568.0) : 1.0)

////////////////////////////////////////////////
/////////////////////////////

CG_INLINE CGFloat
```

```
CGRectGetMinXAutolayout(CGRect rect)
{
    CGFloat x = rect.origin.x * ScaleX_Autolayout;
    return x;
}

CG_INLINE CGFloat
CGRectGetMinYAutolayout(CGRect rect)
{
    CGFloat y = rect.origin.y * ScaleX_Autolayout;
    return y;
}

CG_INLINE CGFloat
CGRectGetWidthAutolayout(CGRect rect)
{
    CGFloat width = rect.size.width * ScaleX_Autolayout;
    return width;
}

CG_INLINE CGFloat
CGRectGetHeightAutolayout(CGRect rect)
{
    CGFloat height = rect.size.height *
ScaleX_Autolayout;
    return height;
}

CG_INLINE CGPoint
CGPointMakeAutolayout(CGFloat x, CGFloat y)
{
    CGPoint point;
    point.x = x * ScaleX_Autolayout;
```

```objc
    point.y = y * ScaleY_Autolayout;

    return point;
}

CG_INLINE CGSize
CGSizeMakeAutolayout(CGFloat width, CGFloat height)
{
    CGSize size;
    size.width = width * ScaleX_Autolayout;
    size.height = height * ScaleY_Autolayout;

    return size;
}

//////////////////////////////////////////////////
/////////////////////////////

CG_INLINE CGRect
CGRectMakeAutolayout(CGFloat x, CGFloat y, CGFloat
width, CGFloat height, BOOL layoutWidth, BOOL
layoutHeight)
{
    CGRect rect;
    rect.origin.x = x;
    rect.origin.y = y;
    rect.size.width = (layoutWidth ? (width *
ScaleX_Autolayout) : width);
    rect.size.height = (layoutHeight ? (height *
ScaleY_Autolayout) : height);

    return rect;
}
```

```
CG_INLINE CGFloat
GetHeightAutolayout(CGFloat height)
{
    CGFloat autoHeight = height * ScaleY_Autolayout;
    return autoHeight;
}

CG_INLINE CGFloat
GetWidthAutolayout(CGFloat width)
{
    CGFloat autoWidth = width * ScaleX_Autolayout;
    return autoWidth;
}

/////////////////////////////////////////////////////
/////////////////////////////
```