

基于opencv下对视频的灰度变换，高斯滤波，canny边缘检测处理，同窗体显示并保存

如题：使用opencv打开摄像头或视频文件，实时显示原始视频，将视频每一帧依次做灰度转换、高斯滤波、canny边缘检测处理（原始视频和这3个中间步骤处理结果分别在一个窗口显示），最后将边缘检测结果保存为一个视频avi文件。

这里问题综合性比较大，这里进行分治。

该类问题可分为四个方面的处理：

- 1. 创建摄像头/播放器
- 2. 将视频文件赋值给播放器
- 3. 逐帧播放
- 4. 释放播放器和文件

以下分为这三个方面进行处理：

(1) 打开 视频或者摄像头，并播放视频

这个利用opencv来说算是非常简单的操作了：

步骤分为四步：（简单操作，播放效果不是最佳的，适用于入门）

- a.创建摄像头/播放器
- b.将视频文件赋值给播放器
- c.逐帧播放
- d.释放播放器和文件

这里直接给上源码（面向对象的方法）：

```
1 // 2 文件名 main.cpp
3 描述 :灰度转换,高斯滤波,canny边缘检测处理视频,并存储canny边缘处理的结果
4 语言 :c++
5 作者 :重交亲爸爸
6 修改 :
7 日期 :2018-05-10
8 说明 :opencv的支持
9 *****
10 #include <cv.h>
11 #include <stdio.h>
12 #include <stdlib.h>
13 #include <iostream>
14 #include<opencv2/highgui/highgui.hpp>
15 #include<opencv2/imgproc/imgproc.hpp>
16 #include<opencv2/core/core.hpp>
17 using namespace cv;
18 using namespace std;
19 *****
20 类名 :MyClass
21 描述 :灰度转换,高斯滤波,canny边缘检测处理视频,并存储canny边缘处理的结果
22 语言 :
23 作者 :
24 修改 :
25 日期 :2018-05-10
26 说明 :
27 *****
28 class MyClass
29 {
30 public:
31     MyClass(char* argv);
32     ~MyClass();
33     void play(); //播放
34 private:
35     CvCapture* capture; //视频播放器
36 };
37 *****
38 函数名称: MyClass
39 函数功能: 初始化
40 传入参数:
41 返回结果:
42 建立时间: 2018-05-10
43 修改时间:
44 建立人:
45 修改人:
46 其它说明:
47 *****
48 MyClass::MyClass(char* argv)
49 {
50     //capture = cvCreateFileCapture(argv[1]);
51     capture = cvCaptureFromFile(argv[1]);
52     if (!cvGetCaptureProperty(capture, CAP_PROP_FRAME_COUNT))
53     {
54         exit(1);
55     }
56 }
57 *****
58 函数名称: ~MyClass
59 函数功能: 释放空间
60 传入参数:
61 返回结果:
62 建立时间: 2018-05-10
63 修改时间:
64 建立人:
65 修改人:
66 其它说明:very important
67 *****
68 MyClass::~MyClass()
69 {
70     cvReleaseCapture(&capture);
71     capture = NULL;
72 }
73 *****
74 函数名称: play
75 函数功能: 播放
76 传入参数:
77 返回结果:
78 建立时间: 2018-05-10
79 修改时间:
80 建立人:
81 修改人:
82 其它说明:
83 *****
84 void MyClass::play()
85 {
86     cvNamedWindow("播放画面", 1);
87     while (true)
88     {
89         frame = cvQueryFrame(capture);
90         if (!frame) break;
91         cvShowImage("播放画面", (frame));
92         char c = cvWaitKey(33);
93         if (c == 27) break; //Esc的编码为27
94     }
95 }
96 int main(int argc, char** argv)
97 {
98     //char* a = "F:\\Pictures\\fcq.avi";
99     //char* b = "F:\\Pictures\\fcq.avi";
100     if (argc > 2)
101     {
102         MyClass myclass = new MyClass(argv[1]);
103         myclass->play();
104         delete myclass;
105     }
106     else
107     {
108         printf("无利用或表示像存储空间");
109         exit(1);
110     }
111     return 0;
112 }
```

公告

昵称: 重交亲爸爸
性别: 4年
粉丝: 6
关注: 0
+加关注

< 2022年3月 >						
日	一	二	三	四	五	六
27	28	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

搜索

我的标签

C++(7)

opencv3.0(6)

ROI(3)

递归(2)

面向对象(1)

Kotlin(1)

Socket(1)

zbar(1)

条形码识别(1)

二维码识别(1)

更多

随笔档案

2019年3月(1)

2018年5月(6)

2018年3月(2)

阅读排行榜

1. 基于opencv3.0(zbar)条形码和二维码的识别与解码(19991)
2. 基于opencv3.0下的运动车辆检测(8511)
3. 在win10下给vs2013配置opencv3.0(6169)
4. 使用opencv调用24*24点阵字库和*16A SCI序库在图片显示文字数字(2317)
5. 基于opencv下对视频的灰度变换，高斯滤波，canny边缘检测处理，同窗体显示并保存(2078)

评论排行榜

1. 基于opencv3.0下的运动车辆检测(5)
2. 算法训练 2的次幂表示(蓝桥杯C++写法)(4)
3. 基于opencv3.0(zbar)条形码和二维码的识别与解码(2)
4. 基于opencv3.0下的人脸检测和检测部分的高斯模糊处理(1)

推荐排行榜

1. 基于opencv3.0(zbar)条形码和二维码的识别与解码(5)
2. 基于opencv3.0下的运动车辆检测(3)
3. 算法训练 2的次幂表示(蓝桥杯C++写法)(1)

最新评论

1. Re:算法训练 2的次幂表示(蓝桥杯C++写法)
而i-1==1{
show(2);
这个不是很理解
-- 那兵
2. Re:算法训练 2的次幂表示(蓝桥杯C++写法)
大佬牛批

使用vs2013生成解决方案后，打开window power shell进行测试，寻找则文件夹，打开相应的可执行文件

```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

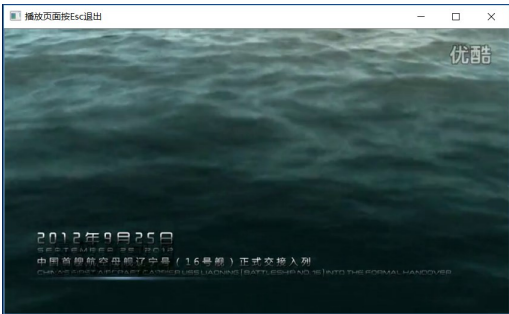
PS C:\Users\ptqp> cd "F:\code\vscode\vscode\capture_show\Debug"
PS F:\code\vscode\vscode\capture_show\Debug> dir

    目录: F:\code\vscode\vscode\capture_show\Debug

Mode                LastWriteTime         Length Name
----                -
-a-----         2018-05-11    16:05           69120 Capture_show.exe
-a-----         2018-05-11    16:05          409108 Capture_show.ilk
-a-----         2018-05-11    16:05          1698816 Capture_show.pdb

PS F:\code\vscode\vscode\capture_show\Debug> .\Capture_show.exe "F:\Pictures\lm.mps"
```

测试结果：



ok 你已经成功完成第一步了。^^

(2) 对视频的每一帧做处理

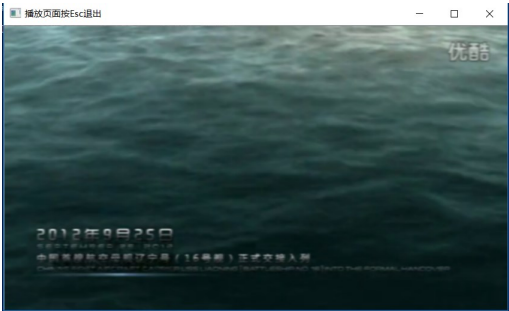
a. 高斯滤波

百度百科的解释：高斯滤波是一种线性平滑滤波，适用于消除高斯噪声，广泛应用于图像处理的减噪过程。通俗的讲，高斯滤波就是对输入图像进行加权平均的过程，每一个像素点的值，都由其本身和邻域内的其他像素值经过加权平均后得到。高斯滤波的具体操作是：用一个模板（或称卷积、掩模）扫描图像中的每一个像素，用模板确定的邻域内像素的加权平均灰度值去替代模板中心像素点的值。

这里有opencv封装好的方法可以直接操作。这里采用的是cvSmooth的方法，然后载入核函数CV_GAUSSIAN，设置图像源和转变后的接收源。

```
1 //*****
2 函数名称: R_Gaussian
3 函数功能: 返回高斯滤波处理
4 传入参数:
5 返 回 值:
6 建立时间: 2018-05-10
7 修改时间:
8 编 写 人:
9 修 改 人:
10 其它说明:
11 *****/
12 IplImage* MyClass::R_Gaussian(IplImage* frame)
13 {
14     //IplImage* edges = cvCreateImage(cvGetSize(image), IPL_DEPTH_8U, image->nChannels);
15     IplImage* edges = cvCreateImage(cvGetSize(frame), frame->depth, frame->nChannels);
16     cvSmooth(frame, edges, CV_GAUSSIAN, 7, frame->nChannels); // Gauss平滑滤波, 核大小为7x7
17     return edges;
18 }
```

按照 (1) 中进行测试，结果：



看上去有点模糊了。

b. 灰度变换

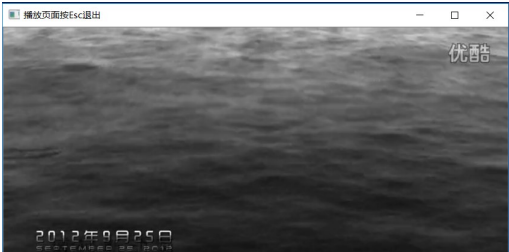
百度百科的解释：灰度变换是指根据某种目标条件按一定变换关系逐点改变源图像中每一个像素灰度值的方法。目的是为了改善画质，使图像的显示效果更加清晰。图像的灰度变换处理是图像增强处理技术中的一种非常基础、直接的空间域图像处理方法，也是图像数字化软件和图像显示软件的一个重要组成部分。

同理，opencv也提供的灰度变化的函数。这里要注意的是转变的时候都是通道变成1

```
1 //*****
2 函数名称: R_Gray
3 函数功能: 返回灰度处理
4 传入参数:
5 返 回 值:
6 建立时间: 2018-05-10
7 修改时间:
8 编 写 人:
9 修 改 人:
10 其它说明: 由于需要在同一个窗体显示, 需要得通道数只收一, 不能使用三通道
11 *****/
12 IplImage* MyClass::R_Gray(IplImage* frame)
13 {
14     //IplImage* edges = cvCreateImage(cvGetSize(image), IPL_DEPTH_8U, 1);
15     //IplImage* frame_gray = cvCreateImage(cvGetSize(image), frame->depth, 1);
16     //IplImage* edges = cvCreateImage(cvGetSize(image), frame->depth, frame->nChannels);
17     cvCvtColor(frame, edges, CV_BGR2GRAY);
18     return edges;
19 }
20
21 IplImage* frame_gray = cvCreateImage(cvGetSize(frame), frame->depth, 1);
22 IplImage* frame1 = cvCreateImage(cvGetSize(frame), frame->depth, frame->nChannels);
23 cvCvtColor(frame, frame_gray, CV_BGR2GRAY);
24 cvCvtColor(frame_gray, frame1, CV_GRAY2BGR); //见说明
25 return frame1;
26 }
```

这里多一步的目的是为了后面多个图像显示在一块窗口上（如果不考虑同窗体显示的话就不需要统一通道数）

结果测试：



- 3. Re-基于opencv3.0下的运动车辆检测
良心文章，感谢，学到了。
--penglaixian
- 4. Re-基于opencv3.0下zbar下条形码和二
维码的识别与解码
请教下博主，angle的值应该是在负90到0
之间，所以检测条形码这段判断代码有什么
用处？新手求教//为了防止找错,要检查这个
矩形的倾斜角度不能超标 //如果超标,那就
是没找到 if (minRect...
--码农翻身做主人
- 5. Re-基于opencv3.0下的人脸识别和识别
部分的高斯模糊处理
这篇也是，人脸检测被当成人脸识别了
--ChrisZZ



灰度效果显示出来了。

canny边缘检测

百度百科的解释: Canny 算法使用 4 个 mask 检测水平、垂直以及对角线方向的边缘。原始图像与每个 mask 所作的卷积都存储起来。对于每个点我们都标识在这个点上的最大值以及生成的边缘的方向。

这里使用cvCanny的方法进行转变。但是这个cvCanny边缘检测还需要弄成灰度图。

```
1 /*****  
2 函数名称: h_Canny  
3 函数功能: 返回Canny边缘处理  
4 传入参数:  
5 返 回 值:  
6 建立时间: 2018-05-10  
7 修改时间:  
8 修 改 人:  
9 修 改 人:  
10 其它说明: 由于我要在同一个窗体显示, 需要开通过数目的, 不能使用单通道  
11 *****/  
12 IplImage* MyClass::h_Canny(IplImage* frame){  
13     //IplImage* edges = cvCreateImage(cvGetSize(grayimage), IPL_DEPTH_8U, 1);  
14     IplImage *frame_gray = cvCreateImage(cvGetSize(frame), frame->depth, 1);  
15     IplImage *frame1 = cvCreateImage(cvGetSize(frame), frame->depth, frame->channels);  
16     cvCanny(frame, frame_gray, 20, 75, 3);  
17     cvCvtColor(frame_gray, frame1, CV_GRAY2BGR);  
18     return frame1;  
19 }
```

同样也要统一通道数。

测试结果:



这样我们三个图像处理就完成了。^^

(3) 同窗体显示四个结果

这里参考了CSDN上的一个方法<https://blog.csdn.net/augusdi/article/details/9019473>

作者稍作整理了下，站在巨人的肩膀上。

```
1 /*****  
2 函数名称: cvShowManyImages  
3 函数功能: 显示多张图片  
4 传入参数: const char* title, int nArgs  
5 返 回 值:  
6 建立时间: 2018-05-10  
7 修改时间:  
8 修 改 人: CSDN Augusdi  
9 修 改 人:  
10 其它说明:  
11 *****/  
12 void MyClass::cvShowManyImages(const char* title, int nArgs, ...){  
13     // img - Used for getting the arguments  
14     IplImage *img;  
15  
16     // DisplayImage - the image in which input images are to be copied  
17     IplImage *displayImage;  
18  
19     int size;  
20     int i;  
21     int w, h;  
22     int w, y;  
23  
24     // w - Maximum number of images in a row  
25     // h - Maximum number of images in a column  
26     int w, h;  
27  
28     // scale - How much we have to resize the image  
29     float scale;  
30     int max;  
31  
32     // If the number of arguments is lesser than 0 or greater than 12  
33     // return without displaying  
34     if (nArgs <= 0) {  
35         printf("Number of arguments too small.....n");  
36         return;  
37     }  
38     else if (nArgs > 12) {  
39         printf("Number of arguments too large.....n");  
40         return;  
41     }  
42     // Determine the size of the image,  
43     // and the number of rows/cols  
44     // from number of arguments  
45     else if (nArgs == 1) {  
46         w = h = 1;  
47         size = 300;  
48     }  
49     else if (nArgs == 2) {  
50         w = 2; h = 1;  
51         size = 300;  
52     }  
53     else if (nArgs == 3 || nArgs == 4) {  
54         w = 2; h = 2;  
55         size = 300;  
56     }  
57     else if (nArgs == 5 || nArgs == 6) {  
58         w = 3; h = 1;  
59         size = 200;  
60     }  
61     else if (nArgs == 7 || nArgs == 8) {  
62         w = 4; h = 2;  
63         size = 200;  
64     }  
65     else {  
66         w = 4; h = 3;  
67         size = 150;  
68     }  
69  
70     // Create a new 3 channel image  
71     displayImage = cvCreateImage(cvSize(100 + size*w, 60 + size*h), 8, 3);  
72  
73     // Used to get the arguments passed  
74     va_list args;  
75     va_start(args, nArgs);  
76  
77     // Loop for nArgs number of arguments  
78     for (i = 0, n = 20; i < nArgs; i++, n -= (20 + size)) {  
79  
80         // Get the Pointer to the IplImage  
81         img = va_arg(args, IplImage*);  
82  
83         // Check whether it is NULL or not  
84         // If it is NULL, release the image, and return  
85         if (img == 0) {  
86             printf("Invalid arguments");  
87             cvReleaseImage(&displayImage);  
88             return;  
89         }  
90  
91         // Find the width and height of the image  
92         x = img->width;  
93         y = img->height;  
94  
95         // Find whether height or width is greater in order to resize the image  
96         max = (x > y) ? x : y;  
97  
98         // Find the scaling factor to resize the image
```

```
199 scale = (float)((float)max / size);
200 // Used to Align the images
201 if (i & w == 0 && m != 20) {
202     m = 20;
203     n += 20 + size;
204 }
205
206
207 // Set the image ROI to display the current image
208 cvSetImageROI(DiapImage, cvRect(m, n, (int)(k / scale), (int)(y / scale)));
209
210 // Resize the input image and copy the it to the Single Big Image
211 cvResize(inp, DiapImage);
212
213 // Reset the ROI in order to display the next image
214 cvResetImageROI(DiapImage);
215 }
216
217 // Create a new window, and show the Single Big Image
218 //cvNamedWindow( title, 1 );
219 cvShowImage(title, DiapImage);
220
221 //cvDestroyWindow(title);
222
223 // End the number of arguments
224 va_end(args);
225
226 // Release the Image Memory
227 cvReleaseImage(&DiapImage);
228 }
```

🔗

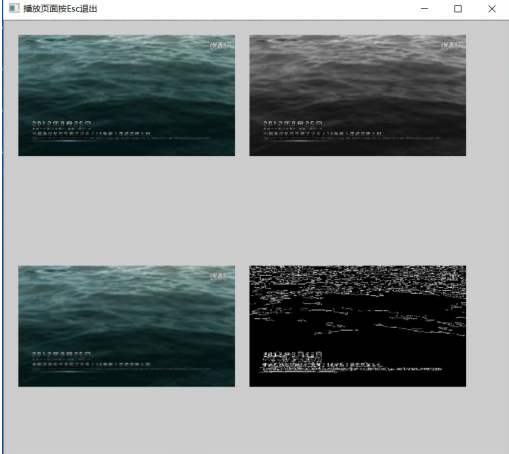
这里加入一个多视频播放处理:

🔗

```
1  /*****
2  函数名称: Multi_play
3  函数功能: 播放
4  传入参数:
5  返回 值:
6  建立时间: 2018-05-10
7  修改时间:
8  建 立 人:
9  修 改 人:
10 其它说明:
11 *****/
12 void MyClass::Multi_play() {
13     IplImage *frame;
14     cvNamedWindow("播放从摄像头Esc退出", 1);
15     cvResizeWindow("播放从摄像头Esc退出", 700, 660);
16     while (true)
17     {
18         frame = cvQueryFrame(capture);
19         if (!frame) break;
20         IplImage* grayImage = I_Gray(frame);
21         IplImage* gainImage = I_Gaussian(frame);
22         IplImage* cImage = I_Canny(frame);
23         //IplImage *frame_not = cvCreateImage(cvGetSize(frame), frame->depth, frame->channels); //另外
24         //cvSetI(frame, frame_not);
25         //cvShowImage("播放从摄像头Esc退出", I_Canny(I_Gray(frame)));
26         //cvWriteFrame(writer, cImage);
27         cvShowManyImages("播放从摄像头Esc退出", 4, frame, grayImage, gainImage, cImage);
28         char c = cvWaitKey(33);
29         if (c == 27) break;
30
31         cvReleaseImage(&grayImage);
32         cvReleaseImage(&gainImage);
33         cvReleaseImage(&cImage);
34         //cvReleaseImage(&frame_not);
35     }
36 }
```

🔗

结果测试:



成功生成, 这里注意一点就是如果放入的图片不是同一渠道的话, 会报错。

(4) 保存视频文件

保存视频就比较简单了。

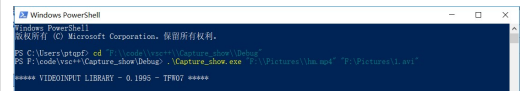
- a.创建视频记录器 **CvVideoWriter* writer;**
- b.初始化视频记录器 如初始化代码所示
- c.视频记录器记录每一帧 如播放功能代码所示
- d.释放视频记录器 如释放代码所示

🔗

```
1  /*****
2  函数名称: MyClass
3  函数功能: 初始化
4  传入参数:
5  返回 值:
6  建立时间: 2018-05-10
7  修改时间:
8  建 立 人:
9  修 改 人:
10 其它说明:
11 *****/
12 MyClass::MyClass(char* argv, char* filepath)
13 {
14     //capture = cvCreateFileCapture(argv[1]);
15     capture = cvCreateFileCapture(argv);
16     if (!cvGetCaptureProperty(capture, CAP_PROP_FRAME_COUNT))
17     {
18         exit(1);
19     }
20     double fps = cvGetCaptureProperty(capture, CV_CAP_PROP_FPS);
21     CvSize size = cvSize(
22         (int)cvGetCaptureProperty(capture, CV_CAP_PROP_FRAME_WIDTH),
23         (int)cvGetCaptureProperty(capture, CV_CAP_PROP_FRAME_HEIGHT)
24     );
25     writer = cvCreateVideoWriter(
26         filepath,
27         CV_FOURCC('M', 'J', 'P', 'G'),
28         fps,
29         size
30     );
31 }
32
33 函数名称: ~MyClass
34 函数功能: 释放空间
35 传入参数:
36 返回 值:
37 建立时间: 2018-05-10
38 修改时间:
39 建 立 人:
40 修 改 人:
41 其它说明:
42 *****/
43 MyClass::~MyClass()
44 {
45     cvReleaseVideoWriter(&writer);
46 }
```

```
46 cvReleaseCapture(&capture);
47 writer = NULL;
48 capture = NULL;
49 }
50 /*****
51 函数名称: Multi_play
52 函数功能: 播放
53 传入参数:
54 备 注:
55 建立时间: 2018-05-10
56 修改时间:
57 建 立 人:
58 修 改 人:
59 其它说明:
60 *****/
61 void MyClass::Multi_play() {
62     IplImage *frame;
63     cvNamedWindow("播放只读源帧to=退出", 1);
64     cvWaitForWindow("播放只读源帧to=退出", 700, 600);
65     while (true)
66     {
67         frame = cvQueryFrame(capture);
68         if (!frame) break;
69         IplImage* grayimage = I_Gray(frame);
70         IplImage* qimage = I_Quantize(frame);
71         IplImage* cimage = I_Canny(frame);
72         IplImage *frame_not = cvCreateImage(cvGetSize(frame), frame->depth, frame->channels);
73         cvNot(frame, frame_not);
74         //cvShowImage("播放只读源帧to=退出", I_Canny(I_Gray(frame)));
75         cvCvtColorFrameToCvIplImage(cimage);
76         cvShowManyImages("播放只读源帧to=退出", 4, frame, grayimage, qimage, cimage);
77         char c = cvWaitKey(33);
78         if (c == 27) break;
79     }
80     cvReleaseImage(&grayimage);
81     cvReleaseImage(&qimage);
82     cvReleaseImage(&cimage);
83     cvReleaseImage(&frame_not);
84 }
85 }
```

ok, 在window power shell输入的时候要加上保存地址了



视频停止或者终止后, 在相应文件下就生成了新文件.avi了

电脑 > D (F:) > 图片



如需要源码请转移至码云: https://gitee.com/qiqibaba/MediaTest/tree/Capture_show进行源码克隆下载

如有问题请留言讨论。转载请注明出处。 谢谢。

标签: C++, opencv3.0, 灰度转换, 高斯滤波, Canny边缘检测

好文置顶

关注我

收藏该文

分享

评论

重文亲爸爸

关注: 0

粉丝: 6

0 推荐

0 反对

加关注

« 上一篇: 使用opencv调用24*24点阵字库和8*16ASCII字库在图片显示文字数字
» 下一篇: 基于opencv3.0下的人脸检测和检测部分的高斯模糊处理

posted @ 2018-05-11 17:51 重文亲爸爸 阅读(2078) 评论(0) 编辑 收藏 举报

刷新评论 刷新页面 返回顶部

登录后才能查看或发表评论。立即 登录 或者 注册 博客首页

推荐阅读:
记一次 dump 文件分析历程
图解 | 从网上和底理解 MySQL 的索引
技术管理进阶 —— 第三个五年, 独立思考与落地实践
平时的工作如何体现一个人的技术深度?
革命性创新, 动画杀手铜 @scroll-timeline



最新资讯:
Oculus 创始人: 扎克伯格玩了我們, 但脸书已经变成 Oculus
截风、断臂、寻路, 微博的艰难一年
Arm裁员千人, 绝地求生还是理性回归?
腾讯股价跌破300港元, 单日暴跌10%
宇宙真的是从一片虚无中产生的吗?
更多新闻...