个人资料

worldpharos

访问： 45866次
积分： 610
等级： BLOG 3
排名： 千里之外

原创： 4篇 转载： 55篇
译文： 1篇 评论： 8条

文章搜索

文章分类

Audio (2)
DirectX (1)
FFMPEG (2)
java (1)
RD (16)
video (7)
嵌入式 (4)
杂谈 (12)
语音编码 (5)

文章存档

2012年06月 (6)
2012年02月 (1)
2011年12月 (1)
2010年12月 (1)
2010年08月 (1)

展开

阅读排行

中国最好的职业TOP10 (6305)
The Scalable Video Codi (2693)
Dual channel/Stereo/Joir (1691)
makefile实例子 (1606)
doxygen使用札记 (1588)

Tutorial: The H.264 Scalable Video Codec (SVC)

分类： video 2008-12-02 17:50 677人阅读 评论(0) 收藏 举报

h.264 video codec stream scalability network

Codecs are used to compress video to reduce the bandwidth required to transport streams, or to reduce the storage space require[illegible]hem. The price for this compression is increased computational requirements: The higher the compression ratio, the more computational power is required.

Fixing the tradeoff between bandwidth and computational requirements has the effect of defining both the minimum channel bandwidth required to carry the encoded stream and the minimum specification of the decoding device. In traditional video systems such as broadcast television, the minimum specification of a decoder (in this case a set-top box) is readily defined.

Today, however, video is used in increasingly diverse applications with a correspondingly diverse set of client devices—from computers viewing Internet video to portable digital assistants (PDAs) and even the humble cell phone. The video streams for these devices are necessarily different.

To be made more compatible with a specific viewing device and channel bandwidth, the video stream must be encoded many times with different settings. Each combination of settings must yield a stream that targets the bandwidth of the channel carrying the stream to the consumer as well as the decode capability of the viewing device. If the original uncompressed stream is unavailable, the encoded stream must first be decoded and then re-encoded with the new settings. This quickly becomes prohibitively expensive.

In an ideal scenario, the video would be encoded only once with a high efficiency codec. The resulting stream would, when decoded, yield the full resolution video. Furthermore, in this ideal scenario, if a lower resolution or bandwidth stream was needed to reach further into the network to target a lower performance device, a small portion of the encoded stream would be sent without any additional processing. This smaller stream would be easier to decode and yield lower resolution video. In this way, the encoded video stream could adapt itself to both the bandwidth of the channel it was required to travel through and to the capabilities of the target device. These are exactly the qualities of a scalable video codec.

H.264 SVC

The Scalable Video Codec extension to the H.264 standard (H.264 SVC) is designed to deliver the benefits described in the preceding ideal scenario. It is based on the H.264 Advanced Video Codec standard (H.264 AVC) and heavily leverages the tools and concepts of the original codec. The encoded stream it generates, however, is scalable: temporally, spatially, and in terms of video quality. That is, it can yield decoded video at different frame rates, resolutions, or quality levels.

The SVC extension introduces a notion not present in the original H.264 AVC codec−that of layers within the encoded stream. A base layer encodes the lowest temporal, spatial, and quality representation of the video stream. Enhancement layers encode additional information that, using the base layer as a starting point, can be used to reconstruct higher quality, resolution, or temporal versions of the video during the decode process. By decoding the base layer and only the subsequent enhancement layers required, a decoder can produce a video stream with certain desired characteristics. Figure 1 shows the layered structure of an H.264 SVC stream. During the encode process, care is taken to encode a particular layer

windows下编译ffmpeg (1578)
西电2012招生计划及近三 (1435)
大恩如大仇---老梁 (1324)
C/C++/Perl/汇编/Java效率 (1226)
H.264 level (944)

评论排行

C/C++/Perl/汇编/Java效率 (3)
高三班主任写给学生的一 (1)
大恩如大仇---老梁 (1)
windows下编译ffmpeg (1)
Dual channel/Stereo/Joir (1)
doxygen使用札记 (1)
MPEG-2中的Profile@Le (0)
H.264 level (0)
指针（详解）本人觉得写 (0)
h.264 profile (0)

推荐文章

最新评论

windows下编译ffmpeg
yeqingbo2010: 我编译的时候遇到了很多问题啊。。。。。头痛！

C/C++/Perl/汇编/Java效率比较
lijunwyf: 兄弟，我有点卖弄了，望多交流

C/C++/Perl/汇编/Java效率比较
lijunwyf: 另外，你知道为什么C偶尔会比汇编快吗？C里面的有些函数用了“寄存器”的数据变量类型，貌似汇编也可以这...

C/C++/Perl/汇编/Java效率比较
lijunwyf: 大哥，你的程序太不公平了1，难道C，C++，就没有异常吗？至少C++就有异常，你应该try,catc...

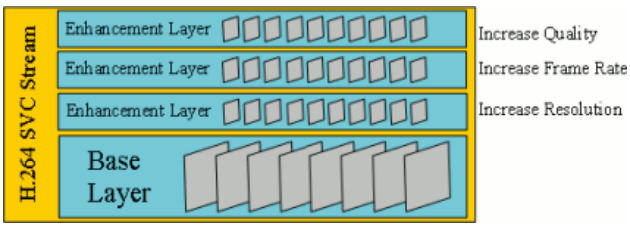
Dual channel/Stereo/Joint stereo
xueyuanyijian: 不错不错，很有用~~

doxygen使用札记
william_ys: 有对C#的语法吗？

大恩如大仇---老梁
匿名用户: 俄罗斯桥很想和老梁交朋友

高三班主任写给学生的一封信（
cherry1982: 路过，随便看看，呵呵。

using reference only to lower level layers. In this way, the encoded stream can be truncated at any arbitrary point and still remain a valid, decodable stream.



(Click to enlarge)

Figure 1. The H.264 SVC Layered Structure.

This layered approach allows the generation of an encoded stream that can be truncated to limit the bandwidth consumed or the decode computational requirements. The truncation process consists simply of extracting the required layers from the encoded video stream with no additional processing on the stream itself. The process can even be performed "in the network". That is, as the video stream transitions from a high bandwidth to a lower bandwidth network (for example, from an Ethernet network to a handheld through a WiFi link), it could be parsed to size the stream for the available bandwidth. In the above example, the stream could be sized for the bandwidth of the wireless link and the decode capabilities of the handheld decoder. Figure 2 shows such an example as a PC forwards a low bandwidth instance of a stream to a mobile device.

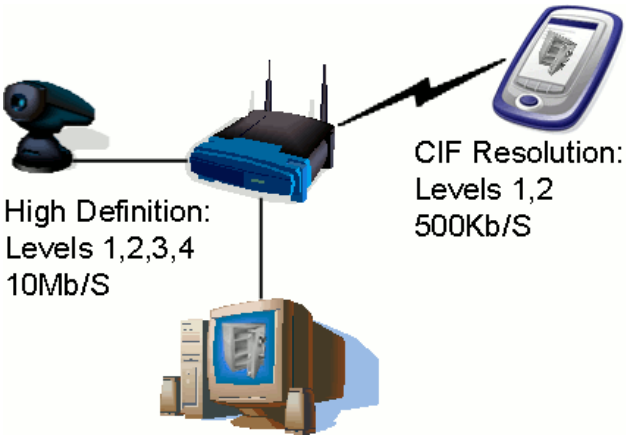
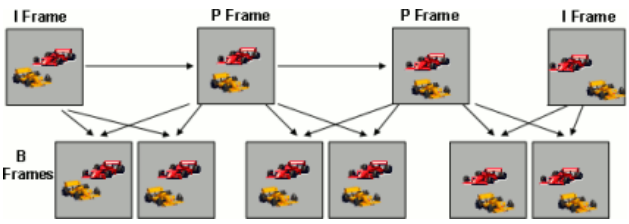


Figure 2. Parsing Levels to Reduce Bandwidth and Resolution.

H.264 SVC Under the Hood

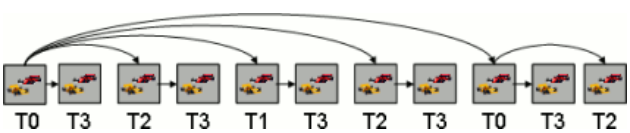
To achieve temporal scalability, H.264 SVC links its reference and predicted frames somewhat differently than conventional H.264 AVC encoders. Instead of the traditional Intra-frame (I frame), Bidirectional (B frame) and Predicted (P frame) relationship shown in Figure 3, SVC uses a hierarchical prediction structure.



(Click to enlarge)

Figure 3. Traditional I, P, and B Frame Relationship.

The hierarchical structure defines the temporal layering of the final stream. Figure 4 illustrates a potential hierarchical structure. In this particular example, frames are only predicted from frames that occur earlier in time. This ensures that the structure exhibits not only temporal scalability but also low latency.



(Click to enlarge)

Figure 4. Hierarchical Predicted Frames in SVC.

This scheme has four nested temporal layers: T0 (the base layer), T1, T2, and T3. Frames constituting the T1 and T2 layers are only predicted from frames in the T0 layer. Frames in the T3 layer are only predicted from frames in the T1 or T2 layers.

To play the encoded stream at 3.75 frames per second (fps), only the frames that constitute T0 need be decoded. All other frames can be discarded. To play the stream at 7.5 fps, the layers making up T0 and T1 are decoded. Frames in layers T2 and T3 can be discarded. Similarly, if frames that constitute T0, T1 and T2 are decoded, the resulting stream will play at 15 fps. If all frames are decoded, the full 30 fps stream is recovered.

By contrast, in H.264 AVC (for Baseline Profile where only unidirectional predicted frames are used), all the frames would need to be decoded irrespective of the desired display rate. To transit to a low bandwidth network, the entire stream would need to be decoded, the unwanted frames discarded, and then re-encoded.

Spatial scalability in H.264 SVC follows a similar principle. In this case, lower resolution frames are encoded as the base layer. Decoded and up-sampled base layer frames are used in the prediction of higher-order layers. Additional information required to reconstitute the detail of the original scene is encoded as a self-contained enhancement layer. In some cases, reusing motion information can further increase encoding efficiency.

Simulcast vs. SVC
There is an overhead associated with the scalability inherent in H.264 SVC. As can be seen in Figure 3, the distance between reference and predicted frames can be longer in time (from T0 to T1 for example) than with the conventional frame structure. In scenes with high motion, this can lead to slightly less efficient compression. There is also an overhead associated with the management of the layered structure in the stream.

Overall, an SVC stream containing three layers of temporal scalability and three layers of spatial scalability might be twenty percent larger than an equivalent H.264 AVC stream of full resolution and full frame rate video with no scalability. If scalability is to be emulated with the H.264 AVC codec, multiple encode streams are required, resulting in a dramatically higher bandwidth requirement or expensive decoding and re-encoding throughout the network.

Additional SVC Benefits
Error Resilience:
Error resilience is traditionally achieved by adding additional information to the stream so errors can be detected and corrected. SVC's layered approach means that a higher level of error detection and correction can be performed on the smaller base layer without adding significant overhead. Applying the same degree of error detection and correction to an AVC stream would require that the entire stream be protected, resulting in a much larger stream. If errors are detected in the SVC stream, the resolution and frame rate can be progressively degraded until, if needed, only the highly protected base layer is used. In this way, degradation under noisy conditions is much more graceful than with H.264 AVC.

Storage Management:
Since an SVC stream or file remains decodable even when truncated, SVC can be employed both during transmission and after the file is stored. Parsing files stored to disk and removing enhancement layers allows file size to be reduced without additional processing on the video stream stored within the file. This would not be possible with an AVC file which necessitates an "all or nothing" approach to disk management.

Content Management:
The SVC stream or file inherently contains lower resolution and frame rate streams. These streams can be used to accelerate the application of video analytics or for cataloging algorithms. The temporal scalability also makes the stream easier to search in fast forward or reverse.

An Application Case Study

A typical application for H.264 SVC is a surveillance system. (Stretch offers market-leading solutions in this area; see its web site for details.) Consider the case where an IP camera is sending a video feed to a control room where it is stored, and basic motion detection analytics are run on the stream. The video feed is viewed at the camera's maximum resolution (1280 x 720) on the control room monitors, and is stored in D1 (720 x 480) resolution to conserve disk space. A first-response team also has access to the stream in the field on mobile terminals within the response vehicle. The resolution of those displays is CIF (352 x 240) and the stream is served at 7 fps.

In an implementation using H.264 AVC, the first likely constraint would be that the camera serves multiple streams. In this example, one at 1280x720 resolution and one at 720 x 480 resolution. This places additional cost within the camera, but allows one stream to be directly recorded at the control room while another is decoded and displayed. Without this feature, an expensive decode, resize, and re-encode step would be needed. The D1 stream is also decoded and resized to CIF resolution to feed the video analytics being run on the stream. The CIF resolution video is temporally decimated to achieve 7 frames per second and re-encoded so that it can be made available to the first-response vehicle over a wireless link. Figure 5 shows a potential implementation of the system using H.264 AVC.

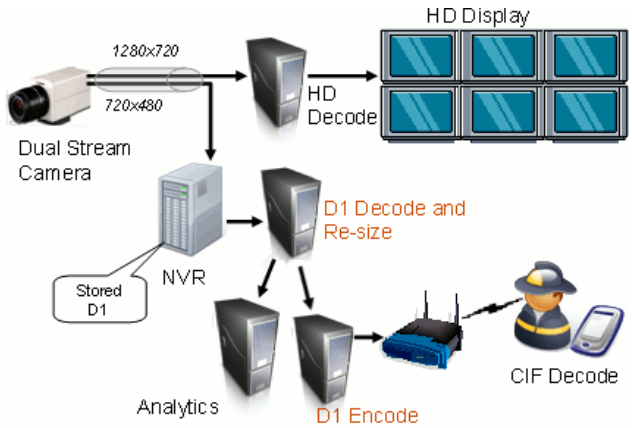


Figure 5. H.264 AVC Video Surveillance Application.

Using an H.264 SVC codec, the multi-stream requirement placed on the camera can be relaxed, reducing complexity and network bandwidth between the camera and control room. The full 1280 x 720 stream can now be stored on the network video recorder (NVR) with the knowledge that it can be easily parsed to create a D1 (or CIF) stream to free up disk space after a specified period. A CIF stream can be served directly from the NVR for analytics work, and a second stream at a reduced frame rate can be made available to the first-response vehicle. Figure 6 shows a potential H.264 SVC implementation.

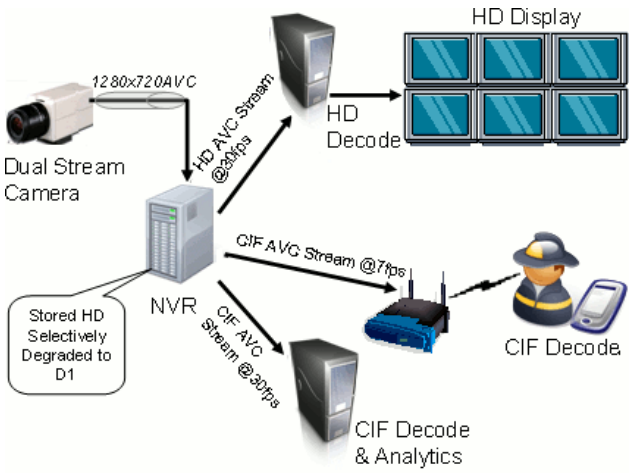


Figure 6. H.264 SVC Video Surveillance Application.

At no point is there a need to operate on the video stream itself—operating on the stored file suffices. The advantages are clear:

- Reduced network bandwidth
- Flexible storage management
- Removal of expensive decode and re-encode stages
- High definition video available on the NVR for archive if needed

Conclusion

Scalable video codecs have been in development for many years. The broadcast industry, strictly controlled by well-established standards, has been slow to adopt this technology. Advances in processor, sensor, and display technology are fuelling an explosion in video adoption. The Internet and IP technologies are seamlessly serving video to an ever more diverse and remote community of display devices. Scalable video codecs such as H.264 SVC satisfy many of the demands of these systems, and they are poised to become a catalyst in wide the spread adoption of video as a communication medium.

About the author

Mark Oliver is the Director of Product Marketing at Stretch. A native of the UK, Oliver gained a degree in Electrical and Electronic Engineering from the University of Leeds. During a ten year tenure with Hewlett Packard, Oliver managed Engineering and Manufacturing functions in HP Divisions both in Europe and the US before heading up Product Marketing and Applications activities at a series of video related startups. Prior to joining Stretch, Oliver managed Marketing for Video and Imaging within the DSP Division of Xilinx.

Related Articles:

- How-to: Next-generation surveillance system design
- How-to: Video Codecs tutorial: Trade-offs with H.264, VC-1 and Other Advanced Codecs
- How-to: Architecting a video transcoding system
- How-to: Evaluation criteria for high-definition video
- How-to: Ensuring high-quality video communications

上一篇 The Scalable Video Coding Amendment of the H.264/AVC Standard
下一篇 Next-generation surveillance system design

查看评论

暂无评论

发表评论

用 户 名: oDavid12345678922

评论内容:

提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack FTC
coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
Angular Cloud Foundry Redis Scala Django Bootstrap