**Lidwine Hô**

France télévisions - innovations & développements

**Hervé Dejardin**

Radio France Innovation

# TUTORIAL

# Audio Production for YouTube 360° videos


November 2016

This tutorial aims to help you produce spatialized audio files that will then be assembled with video files in order to be shown on YouTube 360 °.

In this document we share the method and list of tools we used.

This method seemed to us the simplest to implement.

We used Windows 7 on a PC and the explanations that follow are limited to this operating system. It is possible to reproduce this method with a Mac on OS X, but it is out of scope of the present tutorial.

This tutorial does not explain how to produce 360° images.

This tutorial is therefore not exhaustive.

# Contents

# TUTORIAL

# Audio Production for YouTube 360° Videos

## Overview of the production and distribution chain

The YouTube 360° streaming channel supports downloading and playback of 360° spherical videos.

*360° spherical video implies an interactive 360° picture and sound.*

We tested these 360° videos on computers (Win and OSX) with the Chrome browser as well as on Smartphones and Android tablets with the YouTube application.

With the YouTube app for Android you can watch 360° videos with virtual reality goggles suitable for your Smartphone or with Google Cardboard.

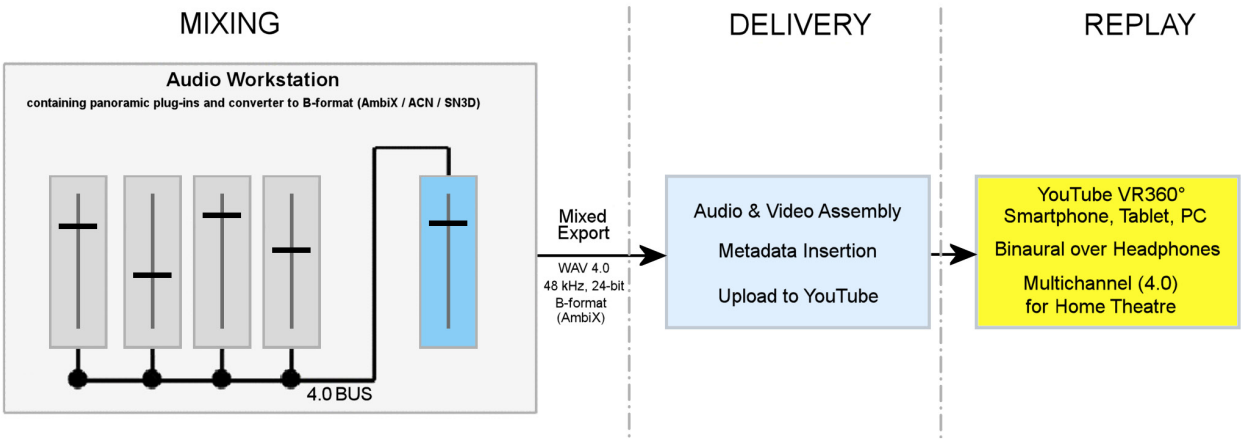
The audio is designed for immersive listening in binaural with headphones. This listening can be interactive and thus simulate the movements of rotation of the head in azimuth (left and right rotation) and in elevation (high and low rotation).

It is also possible to listen to the spatialized audio of YouTube 360° content on 5.1 home theatre. (The audio format used does not allow a great separation of the channels, the quality in 5.1 will not be very good).

For a good quality immersive experience, we recommend binaural listening with headphones and virtual reality goggles.

As described in the following diagram, the audio signal is in a format called Ambisonics. This format allows distribution of spatialized audio with only four channels. It also saves the processor resources needed to simulate rotational movements of the head.

*Note: Currently only the Android YouTube App allows you to listen binaurally.*




**Production chain for YouTube 360° Content**

## Some explanations regarding the reproduction of binaural audio and on the Ambisonics format used by YouTube

### *Binaural Audio*

Binaural audio playback uses HRTFs (Head Related Transfer Functions). HRTFs filter the signal to recreate the complex benchmarks that help us locate the sounds.

Our brain essentially uses three clues to localise itself in space:

- The first clue is the difference in intensity between our two ears. The ear that perceives the most intensity is the ear closest to the audio source.
- The second clue is the difference in the time the sound takes to reach our two ears. The ear closest to the sound source perceives it first. If the sound arrives with the same intensity and at the same moment in our two ears, then the sound source is either in front of us at 0° or behind us at 180°.

We use these first two clues to locate the source on the horizontal plane.

- The third clue is the filtering of the audio source generated by our morphology. Indeed, the spectrum of the sound source is modified by the incidence of our shoulders, our head, our earrings ...

We use this third clue to locate the source on the median plane (front back, top down).

In order to remove the ambiguities of localization, we turn our heads and we constantly try to bring the sounds (or the objects that produce them) into our field of vision.

The quality of the listening experience of binaural audio can be greatly improved with head motion tracking sensors. These sensors are used in virtual reality.

Thus, for listening with headphones, we can reproduce a sound scene in "3D audio" that appears in front of the listener depending on the orientation of his head.

### *Ambisonics and B-format*

We can make an analogy between the Ambisonics format and 360° video: We record the entire scene, but we only look in one direction at a time; we choose our point of view and listening focus in real time.

The Ambisonics format must currently be seen as ubiquitous.

It is a "description" format of a 3D audio scene, which can be decoded in any rendering format; it is very flexible.

First order Ambisonics B-format or FOA (First Order Ambisonic) consists of 4 audio components, referred to as W, X, Y and Z. There are other variants of Ambisonics that use more components. These are termed high order Ambisonics (HOA) and they contain $(order + 1)^2$ components (9 audio components for order 2, 16 for order 3 ...)

The higher the order, the more precise is the "spatial description" of the audio scene.

The Ambisonics format is agnostic to the content that makes up the audio scene, it is a "scene-based" format (based on the description of the entire 3D audio scene).

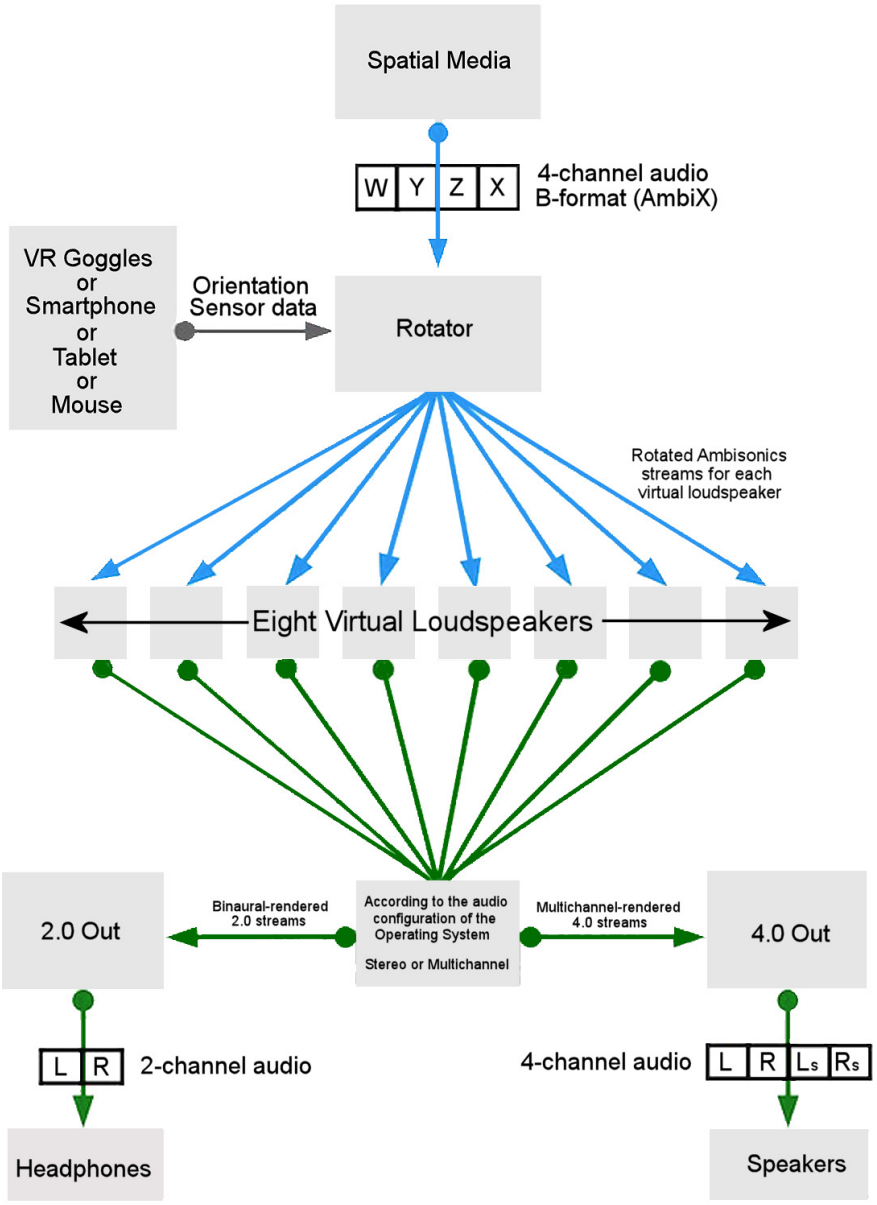
Several of the advantages of the B-format (Ambisonics order 1 or FOA) are that it is not very gourmand in number of signal components needed and that manipulations are quite simple; for instance, the audio scene can very easily be rotated.

The Ambisonics format is not only a pivot or rendering format, it also allows scenes to be

recorded with so-called Ambisonics microphones. For Ambisonics order 1 the different audio channels obtained are in a non-standardized A-format. It is therefore necessary to convert them from A format to B format using a tool generally provided by the manufacturer of the microphone. This tool matrixes and processes the signals produced by the microphone, which typically contains four microphone capsules with cardioid directivity pointing to the apices of a tetrahedron.

The B format allows you to choose not only the direction of listening but also the listening format (mono, stereo, 4.0, 5.1, binaural ...).

It is enough that the "player" knows how to decode the information contained in the B format to convert it to the desired reproduction format.




*Shown is an illustration of the treatment of the Ambisonics signal that allows for rotational movements of the head with an audio output in binaural or multichannel format. This processing chain is similar to that used by YouTube360.*

Many 360° audio manipulation tools employ Ambisonics format as the pivotal format of their render engine.

As the accuracy of rotation is better with higher orders, the processing chains are sometimes in the order 2, 3, 4, ... 7, ...

On YouTube360°, the (order 1) B-format is exclusively used.

## *The two main data exchange formats of B-Format*

There are two main B-Format exchange standards ; FuMa and AmbiX.

It is quite simple to switch from AmbiX to FuMa and vice versa, by means of plugins (see links at the end of the document).The signal components that comprise these Ambisonics exchange formats are audio channels that are intended to be decoded by a calculation of spherical harmonics.

1. The FuMa B format standard uses a channel order and weighting that was proposed by two researchers (Furse and Malham). The B (FuMa) format lists the channels in the order: W, X, Y, Z. The W channel is attenuated by 3 dB.
2. The AmbiX B-format standard has a different channel order and weighting. It lists the channels in the order :W, Y, Z, X and the channels use SN3D normalization (For Ambisonics order 1 it simply means that the four channels have a uniform gain normalization).

Both FaceBook and YouTube use the AmbiX B-format.

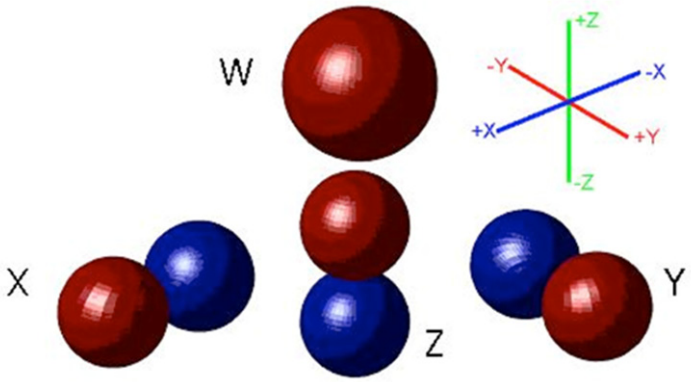
## *Technical Details*

The B-format is based on four audio components, named W, X, Y and Z, as described below.

For a listener at the centre of the sound field, the 3D Ambisonics field is defined on a set of mutually orthogonal axes (X, Y, Z).

By convention, X represents the axis from front to back, Y from left to right and Z from top to bottom. The X, Y, Z channels contain information about the sound field along the corresponding axis (they represent what a bidirectional microphone can pick up along this axis).

The W-channel can be considered as an omnidirectional channel.




## Recording

As previously mentioned, it is possible to directly record in Ambisonics B-format and microphones and recorders for B format are available at reasonable prices.

For YouTube 360° it will be necessary to make sure that the B format is in the AmbiX standard.

Conversion plug-ins from FuMa to AmbiX exist (see end of document).

Recordings that have been acquired in a channel-oriented audio format (mono, stereo, quadraphonic, 5.1 ...) can also be converted to Ambisonics B format.

This conversion can be done with (VST, AAX, AU ...) plug-ins installed on audio mixing and editing software (e.g. Cubase, Pyramix, Reaper, Logic, Pro-tools ...).

## Mixing

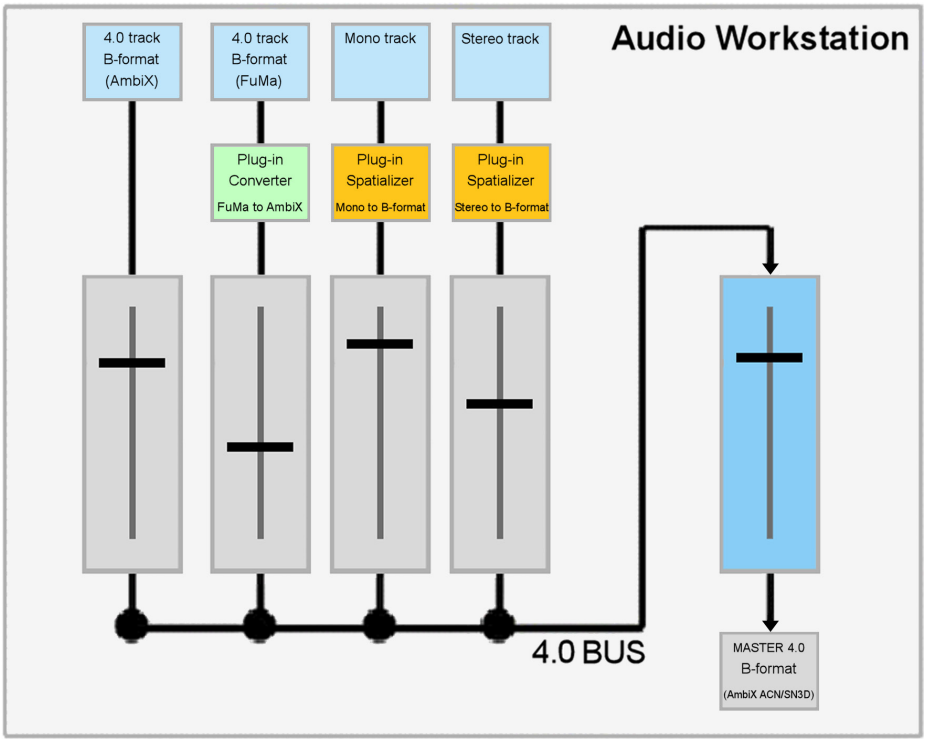
Mixing can be done with any audio mixing and editing software that accommodates four channel buses.

First it will be necessary to create a project with a master bus that must be in a 4.0 multichannel format.

Depending on the format of the audio tracks feeding the master bus, it will be necessary to insert plug-ins to convert or process the signal to B format (AmbiX).

*Here is a simple example of Ambisonics mixing from four different audio source formats (AmbiX, FuMa, mono and stereo).*

*The main (master) output will be in 4.0 B-format. This output will be used to export the mix.*




## Audio Export

Once mixing is complete, the spatialised audio in the B (AmbiX) format is exported.

For optimum quality, it is recommended that the signal is maintained at 48 kHz, 24 bit word length 4.0 interlaced multichannel WAV.

## *Audio levels for export*

It is currently not very easy to predict the listening levels on the player of YouTube 360°.

The level measured on the master 4.0 B (AmbiX) will not be that of the binaural audio output of the player of YouTube360. The processing and (HRTF) filters used by the player during the encoding of the B format into binaural modify the level of the output signal.

Currently, the information necessary for the perfect simulation of the processing chain developed and used by YouTube is not available publicly.

The only (to our knowledge) way to approximate the binaural output level of the YouTube360 player is to simulate it with the AmbiHead plug-in developed by NoiseMakers (see links).

This plug-in works similarly to the Ambisonics-to-binaural encoding found in the YouTube360 player and the few measurements we've made show that the binaural output level of the plug-in is quite close to the binaural output level of the YouTube360 player.

*Note* *This is only a current approximation.*

We recommend a loudness level of -16 LUFS with a level of real peaks at -3 dBTP for the binaural audio output.

## Audio and video assembly

This is to assemble the 48 kHz / 24 bit interlaced 4.0 WAV file (containing B Format AmbiX audio) and an MPEG-4, 16:9 video file with a frame rate of 24, 25, 30, 48, 50 or 60 fps.

The explanations that follow are valid only in Windows (Windows 7 or higher).

*Note: These actions can also be carried out under OSX, but that is out of scope here.*

Assembling the audio and video files for YouTube 360° must be done using the *ffmpeg* library.

To our knowledge, this is currently the only assembly solution that works perfectly.

With other methods we've tried, the software provided by YouTube "refused" to validate the 360°audio option when injecting metadata (see below).

*The success of these new formats will surely motivate developers to produce software that will make this operation easier.*

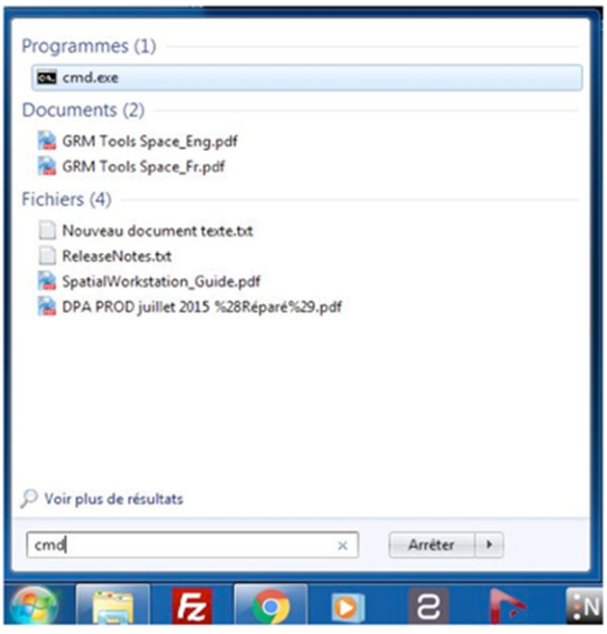
The *ffmpeg* library (Windows version) can be downloaded from:
https://ffmpeg.org/download.html

To install the *ffmpeg* library in Windows, follow the link to the following tutorial:
http://www.wikihow.com/Install-FFmpeg-on-Windows

The following implies that the *ffmpeg* library is properly installed on your computer using the Windows 7 operating system or later.

You must open a command window under Windows.

From the Start menu, type '**cmd**' in the "Search for programs and files" prompt.

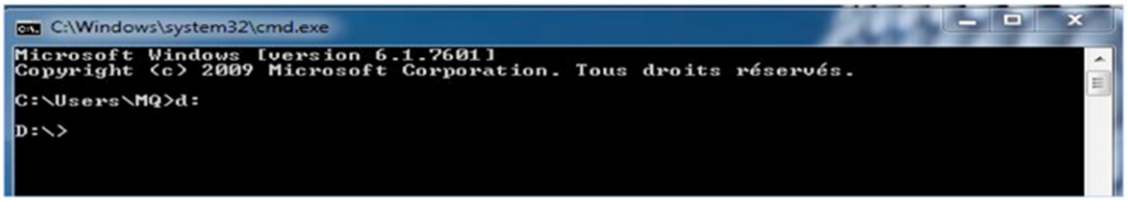



The command window opens and it is through text entry in this window that you will be able to use the *ffmpeg* library for the assembly of video and audio files.

*Tip: use the root of a disk to process files as this will make command line entries much shorter to type.*

For example, for files located in the root of disk D:, enter d: in the command window, press the

‘Enter’ key and then all further *ffmpeg* commands are at file-name level.




The following command line string assembles ‘VIDEO_IN.mp4’ and ‘AUDIO_IN.wav’ into the output ‘VIDEO_OUT.mov’ in the root of the same drive.

```
ffmpeg -i VIDEO_IN.mp4 -i AUDIO_IN.wav -c:v copy -c:a pcm_s24le -af
"pan=4.0|c0=c0|c1=c1|c2=c2|c3=c3" VIDEO_OUT.mov
```


## Metadata Insertion

A final operation is required before uploading the file to the YouTube servers.

Metadatamust be put into the file that will allow the YouTube server to identify whether the file contains a 360° or a stereoscopic image and whether the audio is spatialized.

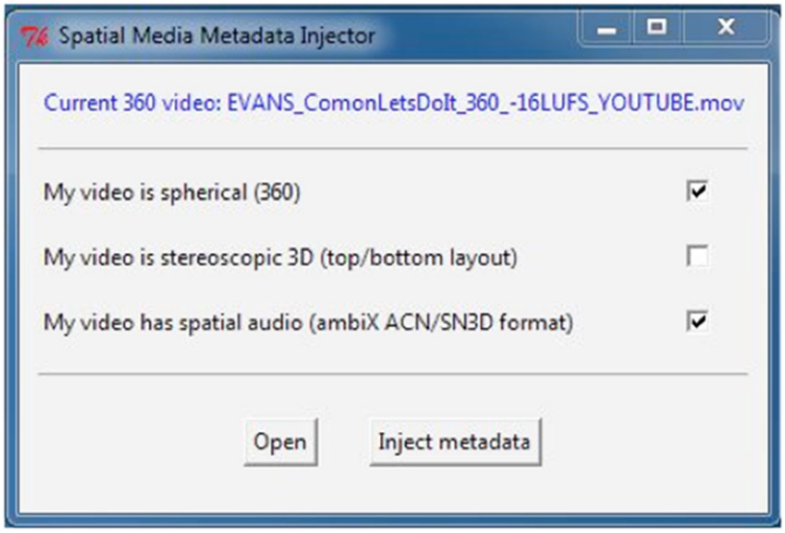
A suitable application is the ‘*Spatial-Media MetaData injector*’, which is free and for windows it is available at this address:

https://github.com/google/spatial-media/releases/download/v2.0/360.Video.Metadata.Tool.win.zip

Once downloaded and unzipped, the application allows you to select the file to process.

Then you have to validate the different options (Spherical 360, Stereoscopic 3D, Spatial Audio).

The following image illustrates the case of a spherical 360° video with spatialized audio.




Confirm by clicking on ‘Inject Metadata’.

The application generates a copy of the selected file with the metadata inserted. It adds the suffix "injected" to the file name.

## Delivery

All you need to do is to upload files to a YouTube account in the usual way.


After the file has been imported by the YouTube server it will take between 15 and 30 minutes before the video is processed into a fully functional 360° video.

## Replay

To be sure to benefit from the sound and the 360° image with interactivity, we recommend:

- Using the YouTube app installed on a recent smartphone or tablet whose OS is Android.
- Using the Chrome browser on a Mac or PC. In this case you can choose between listening in stereo or in 5.1 using loudspeakers. In 5.1, both the image and the sound rotate simultaneously under the action of the mouse.

*Note: Currently, only the Android YouTube App allows you to listen binaurally.*

We have experienced 360° immersion with headphones and Homido virtual reality goggles with a Samsung Galaxy S5 smartphone using the YouTube app to play 360° audio and video content.

The immersion will be total (you will forget your surroundings) with good quality VR goggles and headphones.

Have a good virtual trip!...

# Annex: Command lines and Links

## *ffmpeg command lines*

**Obtain the version of the installed *ffmpeg* library.**

```
Ffmpeg –version
```

**Change encapsulation from .mp4 to .mov**

```
Ffmpeg -i input_file.mp4 -acodec copy -vcodec copy -f mov output_file.mov
```

**Encoding to H264 (420p), encapsulated as a .mov**

```
Ffmpeg -y -probesize 5000000 -i YOUR_INPUT_FILE -c: v libx264 -profile: v main -
vendor ap10 -pix_fmt yuv420p -an YOUR_OUTPUT_FILE.mov
```

**Video file assembly with a 4.0 wav audio file**

```
Ffmpeg -i video_file_in.mov -i audio_file_in.wav -c: v copy -c: a pcm_s24le -af
"pan = 4.0 | c0 = c0 | c1 = c1 | c2 = c2 | c3 = c3" video_audio_file_out.mov
```

## *Links*

**YouTube Help**
Https://support.google.com/youtube/topic/2888648?hl=en&ref_topic=16547

**(YouTube Help) Post a 360 ° video on YouTube**
Https://support.google.com/youtube/answer/6178631?hl=en&ref_topic=2888648

**(YouTube Help) Spatialized audio for 360 ° video on YouTube**
Https://support.google.com/youtube/answer/6395969?hl=en&ref_topic=2888648

**Github Google**
Https://github.com/google/spatial-media

**Format Ambisonic**

Https://en.wikipedia.org/wiki/Ambisonics

**Download ffmpeg**

Https://ffmpeg.org/download.html

**Installing ffmpeg on Windows**

http://www.wikihow.com/Install-FFmpeg-on-Windows

**ffmpeg documentation**

Http://ffmpeg.org/documentation.html

**Spatial Media MetaData Injector**

**For Win:**

Https://github.com/google/spatial-media/releases/download/v2.0/360.Video.Metadata.Tool.win.zip

**For OSX:**

Https://github.com/google/spatial-media/releases/download/v2.0/360.Video.Metadata.Tool.mac.zip

**Spatial WorkStation Facebook**

Https://facebook360.fb.com/spatial-workstation/

**Bruce Wiggins Blog**

Http://www.brucewiggins.co.uk/

**Ambisonic Plug-in**

**AmbiX Collection**

Http://www.matthiaskronlachner.com/?p=2015

**NoiseMakers**

Http://www.noisemakers.fr/