

## **Shot List**

### **1. Amazing Spiderman 2 (2014):**

- Reduced render times by 70%: Artist had setting at 12aa at 4k renders to remove noise from and erroneous mesh light artifact in the scene. After removing unnecessary objects, final render settings were at 8aa at 2k saving 70% render time.

### **2. Amazing Spiderman 2 (2014):**

- Created archiving system to reduce scene build time: With the entire times square city of buildings to create for each shot, Ass Archiving system was developed to help reduce scene build times of 60+ min and memory consumption. Final renders were reduced to ~15min scene build times less than 20Gb accounted memory.

### **3. Amazing Spiderman 2 (2014):**

- Created archiving system to reduce scene build time: Due to high memory consumption, the Ass Archiving system was used throughout the sequence for high memory frames with large scene build times. The entire scene was completely covered with pebbles from FX causing high memory and +60min scene build times.

### **4. Amazing Spiderman 2 (2014):**

- Reduced render times by 40%: Spiderman's webs had a continuous issue with noise in the specular layer. I contributed to the reduction of render time in multiple shots in the sequence.

### **5. Smurfs (2013) : McDonald's Commercial Spot**

- Lighting and Precomp: I was responsible for the final lighting and precomp of the two shots. My tasks included the look development of the apple bitten by Hackus along with the integration of the apples in the bag.

### **6. Casteel (2010) : Commerical Spots**

- Layout, Texturing, Tracking, Lighting, Compositing: I was lead production artist on the 3 spot project. Controlling all of backend responsibilities along with capturing the set digitally with track markers and spheron images. Completed using Maya, Shake, PFTrack and After Effects.

### **7. 350z (2009):**

- This was a senior project where I was responsible for the full production. The goal was to integrate a CG element with a live action plate. This was accomplished using Maya, Houdini and Shake with PFTrack to track.