Lean Angle Visualization

Bruce Ng and Marco Cheng

Fig. 1. Pata Yamaha turning around a corner at Circuito de Jerez.

Abstract — This project explores the concept of lean angle, which is one of the most unique aspects of driving a motorcycle. Currently, information about lean angle is scattered around the Internet in various articles and videos. This makes it hard for an interested motorist to learn about the intricacies of riding a motorcycle. As result, the project synthesizes these different articles and videos by creating an interactive visualization that allows motorists to visualize a turn on the motorcycle. The goal of the project is to create a visualization tool that can be used in helping motorcyclist understand the physics behind lean angle. By the understanding the physics behind lean angle, the motorcyclists can apply this knowledge to safely complete turns. In the end, our projects aims to help reduce the amount of accidents that motorcyclists will be involved in due their increased knowledge about the physics of a motorcycle.

Index Terms — Lean Angle, Countersteering, Motorcycle, Speed, Turn Radius, Friction and Circuito de Jere

Introduction

Driving a motorcycle requires a separate motor license because it adheres to different physics principles than a normal passenger car. For example, fake forces and torque allow a motorcycle to a lean significantly during a turn. As result, motorcyclists must learn new driving principles in order to adapt to different physics principles. One phenomenon is called counter-steering. By using handlebar inputs, a motorcyclist initializes a lean that allows the motorcycle to successfully complete a turn. The handlebar input corresponds to the direction of the turn. However, the motorcyclist would lean in the direction opposite to the turn. For example, during a right turn, the motorcyclist would initiate the lean by pushing the handlebars to the right. However, in order to balance the motorcycle, the motorcyclist would have to lean left. In terms of a physics explanation, the motorcyclist must shift the center of mass in order to change the inertia of the motorcycle's movement. Inertia comes from Newton's First Law that describes that an object in motion will continue at the same speed and angle until it is acted upon by another force. In other words, inertia describes an object's tendency to resist changes in its motion. As result, the lean induced by the handlebar input and the shift in center of mass causes an unbalanced force to be acted upon the motorcycle. One thing that makes this concept hard to understand is the amount of lean the motorcyclist must apply. The amount of lean, lean angle, must be accurate because the two wheel design of a motorcycle creates unbalanced forces in terms of torque. Torque is the amount of force required for an object to rotate around a center point. If a greater lean angle is applied, then the motorcyclist risks losing control of the motorcycle since the motorcycle's wheels are unable to provide enough grip. On the other hand, if a lower lean angle is applied, then the motorcycle's inertia prevents the motorcycle from changing directions. This can be a safety risk because the motorcyclist risks crossing into another lane of traffic during the turn. As result, it is important for motorcyclist to have a physics background so they can visualize the lean angle necessary to safely complete a turn.

Related Works

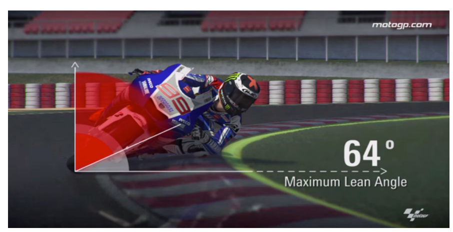

Fig 2. Video from MotoGP

When searching for information about lean angle, it was hard for us to find a comprehensive source. Our initial source of information came from a video made by the MotoGP to illustrate the impressive engineering feat of their motorcycles. The video did a great job in providing a visual representation of lean angle. For example, it showed that a lean angle of 64 degrees was dramatically different from a lean angle of 40 degrees since 64 degrees was far closer towards the ground. However, one of the major weaknesses of the video was that it didn't provide any information about the physics behind lean angle. As result, it didn't help a motorcyclists understand the factors that influence lean angle. On the other hand, we found an article by Rhett Allain from WIRED that focused on describing the physics behind lean angle. Using a free body diagram, Allain was able to generate two net force equations to solve for the lean angle.

Afterwards, Allain simplified the equation by showing how different variables would cancel themselves. In the end, Allain was able to provide a simple system of equations that related turn radius, speed and lean angle While this article was very helpful in understanding the physics behind lean angle, it was a bit hard to understand because it expected a familiarity with physics. For example, it used lots of physics concepts like torque, curvature and fake forces. In addition, it only provided two examples where real numbers were used to illustrate how the simplified system of equations can be used to calculate the lean angle. However, both examples stopped at just calculating the friction used in the lean angle formula. In addition, the article didn't help the user visualize the calculations. For example, radius of turn can be hard to visualize a turn because we are used to using angles for turns. As result, when we looked carefully at our two main sources of related works, we found a gap that our exploration could explore. We wanted to combine the complex physics formula described in Allain's article with the visual representations of lean angle seen in the MotoGP video. As result, our goal was to allow the user to create visual representations of lean angle by manipulating the speed and turn radius of a motorcycle. This would help the user learn more about lean angle without necessitating the deeper knowledge of physics used in the Allain article.

METHODS

After deciding upon the goal of Lean Angle Calculator, the next step in the process was to develop it. The first task for us was to decide the libraries that we would use to create the visualization. We were quick to agree in exploring Three.js because it's a JavaScript library that simplifies the process of creating animated 3D graphics. We knew that we wanted to use 3D graphics because it would show the difference between different degrees of lean angle. In addition, it would help the user visualize the use of turn radius, rather than angle of the turn, to describe the turn. Once we have decided upon the libraries, the next task was to create our storyboard.

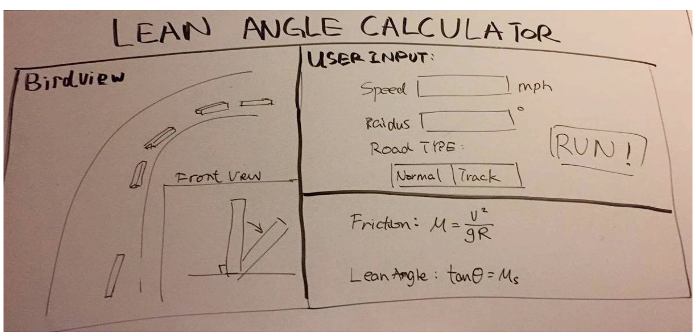

Fig 3. Final Storyboard

One of the greatest challenges that we faced when we were developing our storyboard was the scope of our project. Since neither of us have experience in using Three.js, we weren't too knowledgeable about the limitations or complexity of the library. As result, we were worried about implementing features where Three.js didn't provide great documentation. In the end, we decided to implement three features in our visualization. The first feature was the user input section where the user can manipulate speed and turn radius in order to explore how they impact lean angle. This provides the interactive element to our visualization since the user can manipulate the simulations created in Three.js. The second feature is the front view of the motorcycle. It would use Three.js to provide a animation where a motorcycle would come from a standing up position (0 degree lean angle) to the calculated lean angle. This allows the user to see how much the motorcycle must lean in order to successfully complete the lean. The last feature that we wanted to implement was a track with the motorcycle on it. This allows the user to visualize the tightness of the turn based upon the turn radius since they would follow the motorcycle as it makes the turn. This would directly address the problem of using turn radius, rather than angle to describe a turn. Through further research, we found that Three.js had physics library called Physijs that could handle movement of the motorcycle.

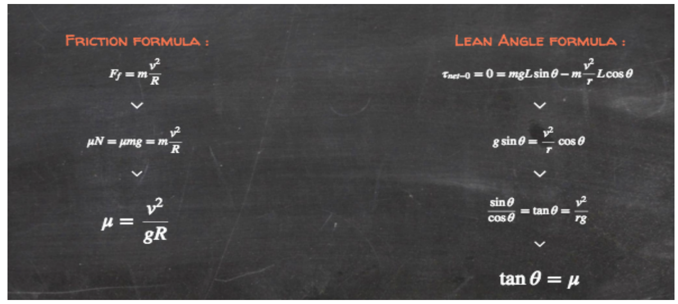

Fig 4. Deriving the Physics Formula

Once we have finalized the features in the storyboard, the next important step was to understand the physics behind calculating lean angle. As result, the main goal was to understand the implications of the system of equations that Allain provided. From the friction formula, the first thing that we noticed was that friction would have no units. This is apparent since g meant gravity, which was consistent at 9.81 meters per second squared. In Allain's article, he described his use of speed and turn radius through meters per second and meters. Inputting these units into the friction showed that the meters and second units would cancel out. Since friction had no units, it means that we would have worry about two different conversions. The first conversion was between radians and degrees. While lean angle was calculated in degrees, we noticed that JavaScript's Math library uses only radians in its calculation. This means we had many opportunities for error in calculating lean angle if we didn't track our units. The second units is that we have to calculate speed in meters per second and turn radius in meters so we can ensure that the units will cancel out for friction. This proved to be a challenge for us since we were used to describing distance through feet and speed through miles per hour. However, if we decided to use familiar units, it would mean that we would have to do lengthy conversions that harmed the preciseness of our calculations. In the end, we decided to continue with using feet and miles per hour to describe our turn radius and speed because it was easier for us to visualize what those numbers meant.

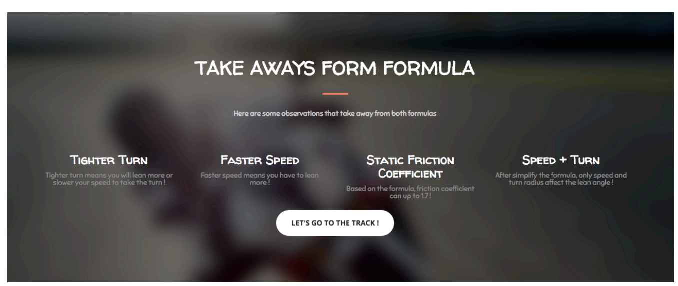

Fig 5. Formula Takeaway Summaries

In addition to conversions, there were other implications that we learned from the physics formula. The first thing that we learned is that a faster speed creates a bigger lean angle. This is apparent with the velocity squared in the friction formula. This means that speed will impact the friction equation more than turn radius. The second thing that we learned is that the coefficient of friction can reach 1.7. Currently, the maximum lean angle that a MotoGP motorcycle can support is 64 degrees. Using the lean angle formula, this means that the friction must be 1.7. This was confusing for us because we initially thought that friction must be between 0 and 1. This was useful for our last graph (friction v. lean angle) because we know that there is no upper bound of 1 for the coefficient of friction. The final thing that we learned is that a tighter turns required a bigger lean angle or a slower speed in order to safely make the turn. Since turn radius is the divisor in the end, a decrease in turn radius would increase the dividend (speed). As result, speed would have to equally decrease in order to keep the same friction calculated from turn radius and speed. On the other hand, if the motorcyclist decides not to decrease speed, then they have to increase lean angle to compensate.

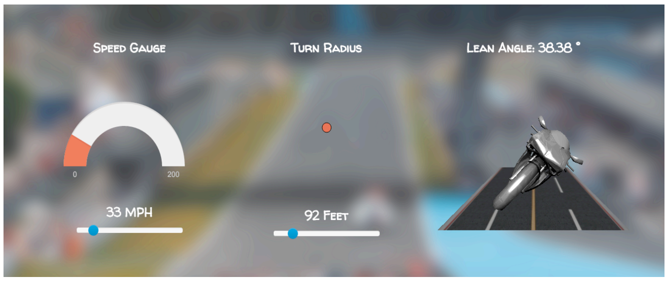

Fig 6. User inputting speed and turn radius to calculate lean angle

With the two conversions in the back of our mind, we were able to calculate lean angle from user inputs in feet and miles per hour. As result, the next step was to utilize Three.js to create the 3D animations. The first challenge that we faced was finding a motorcycle model that we could use since Three.js only provides the option of generating meshes that are based off geometric shapes (i.e. cubes and spheres). We solved this problem by finding a motorcycle on Clara.io, which is a website that allows 3D models to be exported into Three.js's ObjectLoader function through a JSON format. The next challenge was writing a formula that rotated the motorcycle at a constant speed. One of our biggest worries is that small lean angles (i.e. 15 degrees) would animate quickly, meaning that it might be difficult to see the change in lean angles. As result, we explored how the render function of Three.js works. We discovered that the render function of Three.js has a fixed time step of 1/120. This means that the render function will be called 120 times every second. This helps make the animations feel smooth by updating it multiple times per a second. Knowing that we had to use radians in JavaScript, the next thing that we did was calculate the lean angle in radians because it would show how much our motorcycle must be rotated. In order to animate the lean angle at roughly the same time, we decided to rotate the motorcycle by 1/120 of the target radian every frame until the motorcycle had rotated to the calculated lean angle. Our goal was to have the animation be completed in roughly a couple of second.

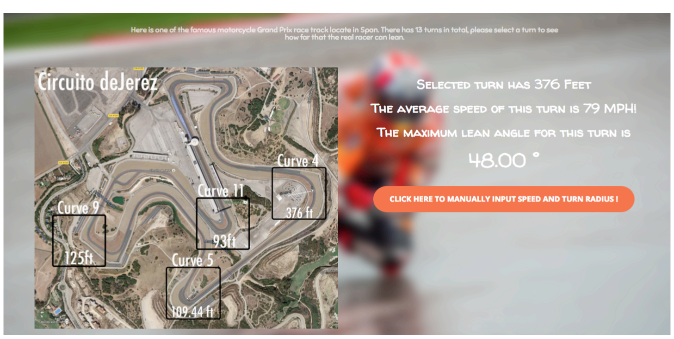

Fig. 7. Circuito de Jerez Track

Once we had implemented the front view animation and the user input section from our storyboard, the final task was to implement the track with the motorcycle on it. However, we faced difficulties in generating a curved track in Physijs. In addition, our turns with the motorcycle were very inconsistent. Since we were fearful of the many potential edge cases that we didn't design for, we decided to go towards another direction in order to help the user visualize turn radius. We found a picture of a motorsport track where MotoGP competitions occurred. The track would provide pre-generated information about the average speed and turn radius at the turns. Specifically, we picked Circuito de Jerez because we found a speed map that shows the average speed entering each turn. However, finding the turn radius was harder because we couldn't find this information online. As result, we improvised by using Google Maps. By placing two points at the beginning and end of a turn, Google Maps could provide an estimate of the turn radius. However, since this was a very rough way of estimating because we didn't use any pre-defined rules or formulas that ensured consistency. As the user clicks on these turns on the Circuito de Jerez, it would automatically enter the speed and turn radius into the user input section. This also serves as a tutorial for the user because it provides sample speed and turn radius numbers for the user so they know what numbers are realistic to use. This was important for us because we were worried that our visualization tool might be challenging to use since many information was based on physics.

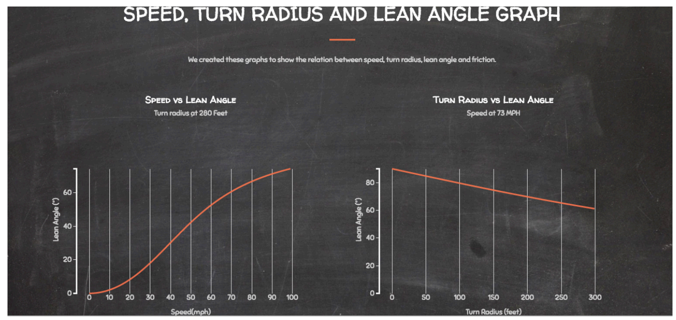

Fig 8. Speed v. Lean Angle and Turn Radius v. Lean Angle Graphs

Finally, we decided to add another feature that wasn't included in our storyboard through graphs. These graphs were created in d3. Since lean angle depended on the variables of speed and turn radius, we wanted to provide an easy way for the user to see what would happen if they change some elements of the turn. For example, if the user went to a turn slower than the inputted speed, the user can see how it impacts lean angle. In order to simply the graphs, we decided to use two variable graphs. As result, in the speed v. lean angle, we assume that turn radius will always be kept consistent. On the other hand, in turn radius v. lean angle, we assume that speed will be kept consistent. These graphs would also dynamically update based on the user's current speed and turn radius input.

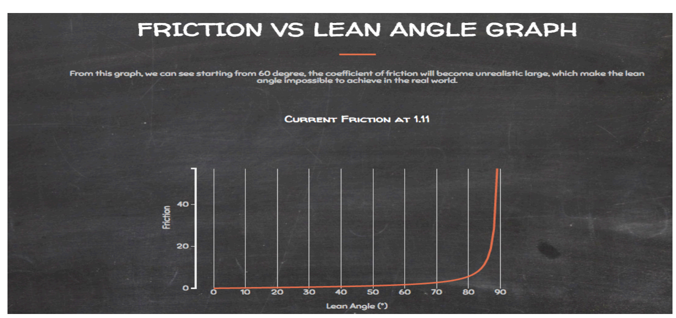

Fig 9. Friction v. Lean Angle Graph

Finally, we made a third graph (friction v. lean angle) that connects the two separate equations of friction and lean angle. The main difference in that this graph is static. Despite the static nature, it is useful because it allows the user to see lean angle formula in a more visual way. Since an inverse tangent graph may be hard for some one to visualize, we thought that it was important that the user could see this relationship. However, since this is one element of lean angle that is targeted toward someone who is interested in the physics component, we decided to put it in the very last section so it is optional for other users.

RESULTS

In the end, our visualization accepts two user inputs to create two different visualizations. The two user inputs are the speed (mph) and turn radius (feet) inputs. With these inputs, the first visualization is the front view of the motorcycle that shows the lean angle that is calculated from the two user inputs. This visualization requires the user's input through the "simulate button". We decided on the feature because dynamically updating the simulation creates a lot of bugs since the user can continually change the sliders before the animation finishes. For example, if the user changes speed inputs quickly, the motorcycle will be rotating infinitely. The main way that we evaluated the success of this animation is through the time it takes to go from 0 degrees to the target lean angle. We wanted the animation to be slow enough so it becomes apparent that the motorcycle was rotating. On the other hand, we didn't the animation to be too slow, causing the user to be disengaged. We wanted this animation to provide an engaging way to demonstrate lean angle. As result, by comparing the animation time to the average time the user spends between clicking the "simulate button", we can determine if the user is spending too much time waiting for the animation to finish loading. On the other hand, the second form for visualization is the graphs that show the relationship between lean angle, speed and turn radius. Since the graphs dynamically update based on the speed and turn radius inputs, the user doesn't need to interact with the "simulate button" to update the visualization. As result, the criterion for evaluating our second approach is the average time between user interactions with the "simulate button". If the average time between "simulate button" is low, it shows that users may not be spending a lot of time looking at contents of the graph. On the other hand, if the average time between "simulate button" is high, it shows that the user may be manipulating speed or turn radius to see how it impacts the lean angle through the graph.

DISCUSSION

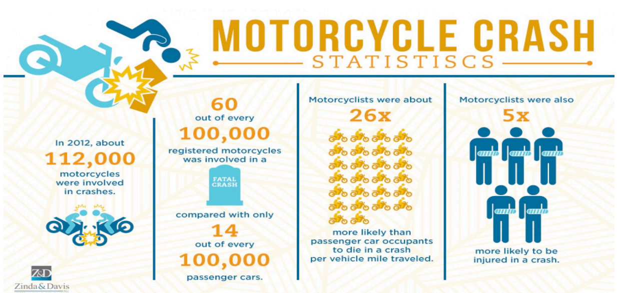

Fig 10: Motorcycle Safety Statistics Compared to Passenger Cars

One of the most alarming statistics about motorcyclists is the safety statistics compared to regular drivers. Motorcycles were significantly more likely to be involved in a crash than passenger cars. The infographic from Figure 9 shows some of the statistics about Motorcycle Crash Statistics. One way to make driving a motorcycle safer is to help the motorcyclist gain a greater understanding of the physics behind their motorcycle. This is important because a motorcycle adheres to different physics principles than a normal passenger car. In addition, motorcyclists have little protection if an accident were to occur, increasing the likelihood of serious injuries. One example where a greater understanding of physics could be useful is addressing the leading cause of motorcycle accidents in speeding. As shown by the physics formula, greater speeds require a greater lean angle in order to successfully complete the turn. However, such physics information may not be taught during the motorcycle training classes. As result, our visualization teaches about the factors that influence how a motorcycle reacts to a turn. For example, a tighter turn would require a slower speed or a greater turn angle in order to compensate. By providing interactive visualizations that help the user see how speed and turn radius impacts lean angle, the user can gain a mental image of lean angle. This mental image can help users gain a better physics understanding that can be applied when they make theses turns on the motorcycle itself.

FUTURE WORKS

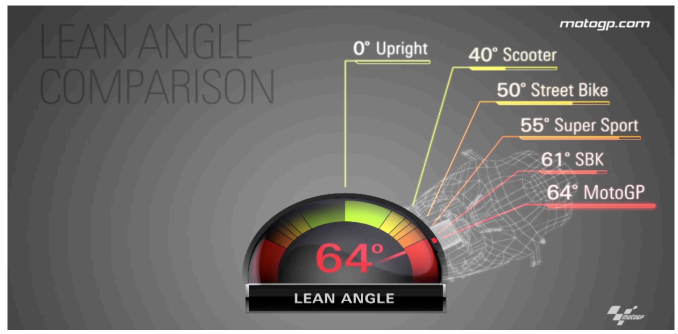

Fig 11: This chart shows the lean angle of different type of motorcycles

Currently, one of the biggest assumptions that our systems make is that the design of the motorcycle doesn't matter. This suggests that all motorcycles have the same lean angle during a time. However, this is not true. For example, sponsors of the MotoGP have spent millions of dollar on researching and developing new motorcycle frames. This allows the motorcycles used in the MotoGP competitions to have lean angle beyond those available to the public. As result, MotoGP motorcycles can safely reach an impressive lean angle of 64 degrees. On the other hand, street motorcycles that are available to the public only have a lean angle of 50 degrees. In the future, we hope to do more research upon the factors of a motorcycle frame that changes lean angle. This would allow the motorcyclist to customize the visualization to the type of motorcycle they are interested in riding. In addition, it would also allow us to provide feedback to the user. For example, in our current animation, we could show the user if certain types of motorcycle were capable of making the currently calculated lean angle. This could help the user learn about the physical limitations of different motorcycle designs.

Static Coefficient of Friction

Road conditions	Normal tire (treaded)	Racing tire (smooth)
dry	0.7	0.9
wet	0.4	0.1
snowy	0.2	
icy	0.1	

Fig 12: Table to show how road conditions impact friction

The second major assumption that we're making in our system is that the road and tire conditions were negligible. In other words, we assume that our calculations were done in an environment where the roads were calm, clear, dry and flat. In addition, we assume that the tires are in fairly new condition. This means that they have no started to lose their grip onto the roads. This is important because these conditions can impact the calculation for lean angle since other factors than speed and turn radius can change the coefficient of friction. For example, a dry road can have a friction coefficient of 0.7 to 0.9. On the other hand, a wet road can have a friction coefficient of 0.1 to 0.4. The effects of road and tire conditions can be seen in the stopping distance of a vehicle. The lower the coefficient of friction, the longer it takes for a vehicle to stop. As result, if a car wants to have the same stopping distance in a wet road, they must enter the stop with a slower speed. This idea can be applied to a motorcycle since the motorcyclist would require a tighter turn or higher speed in order to achieve the same lean angle in wet road conditions,

Fig 13: Example 3D Motorcycle Simulation Software

Finally, the way that we could extend our system is to add a game component to the visualization. One of the main goals of the visualization was to help motorcyclist learn about the physics of a turn. One way that we can strongly improve the learning aspects is through increased user interactions. Currently, the only way that the user may interact with our system is changing the speed (mph) and turn radius (feet). From changing these two variables, the motorcyclist can only see what the motorcycle looks like from an outside view. As result, it might be harder for a new motorcyclist to visualize the lean view while riding the motorcycle. A simulation provided from the first person view would help address that potential problem. When the user clicks on one of the turns from the Circuito de Jerez map, it would create a turn that the user would be responsible for making. The speed of the motorcycle would be taken from the average speeds from the turn. The user would control the motorcycle as it makes the turn. In the corner, it would show the current and target lean angle of the motorcycle so the user knows if they have reached the required lean angle to make the turn. Finally, we envisioned that the user could generate custom tracks by changing the turn radius. This allows the user to practice various types of turns.

References

Allain, Rhett. (2015, September 2). “Just How Far Can A Motorcycle Lean In A Turn?” *Wired*. Retrieved from http://www.wired.com/2015/09/just-far-can-motorcycle-lean-turn/.

“Friction.” (2009, April 11). *University of Massachusetts*. Retrieved from http://blogs.umass.edu/ndarnton/2009/04/11/friction/.

“Infographic: Motorcycle Crash Statistics.” (2012). *Zinda & Davis*. Retrieved from http://www.zdfirm.com/wp-content/uploads/2015/07/motorcycle-crash-statistics.jpg.

Jones, Scott. (2015, May 2). “Jerez Track Map with Speed and Gear Telemetry.” *PhotoGP*. Retrieved from http://photo.gp/2015/05/02/jerez-track-map-with-speed-and-gear-telemetry/.

“MotoGP Lean Angle Experience.” (2013, September 27). *Video*. Retrieved from https://www.youtube.com/watch?v=J73XRDGPcpE&feature=youtu.be&t=53s.

“Motorcycle Simulation.” *(n.d.). Softonic. Retrieved from* http://motorcycle-simulator.en.softonic.com/.

“Pata Yamaha Wrap Pre-Season WSBK Jerez Test.” (2016, January 29). *Motorcycle USA*. Retrieved from http://www.motorcycle-usa.com/2016/01/article/pata-yamaha-wraps-pre-season-wsbk-jerez-test/.

Spiegel, Bernt. (2014, February 6). “Team Arizona Racing Tip: Motorcycle Lean Angle… How Low Can You Go?” *Team Arizona*. Retrieved from http://www.motorcycletraining.com/team-arizona-riding-tip-motorcycle-lean-angle-how-low-can-you-go/.