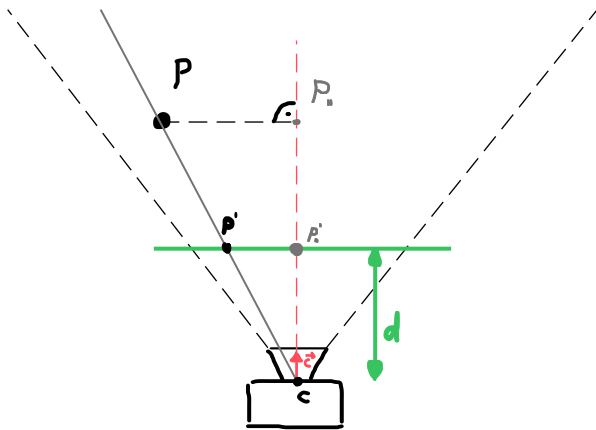


Um Perspektive zu behalten muss P um c zentrisch mit Faktor k gestreckt werden, sodass P immer auf der **ViewPlane** der Kamera landet

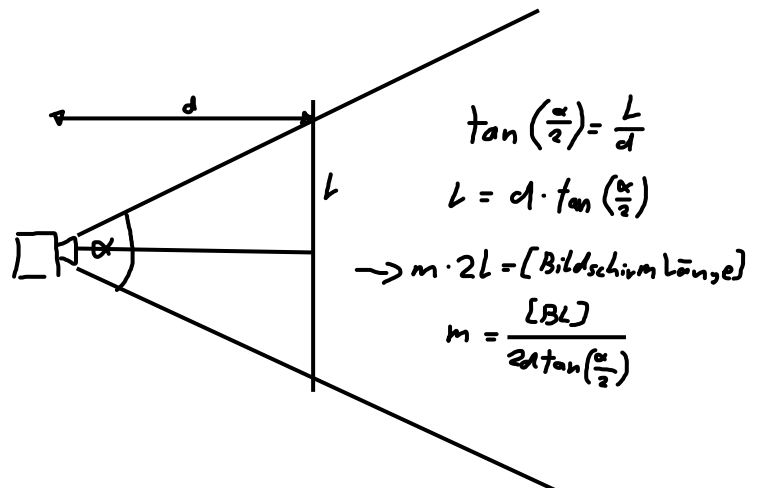
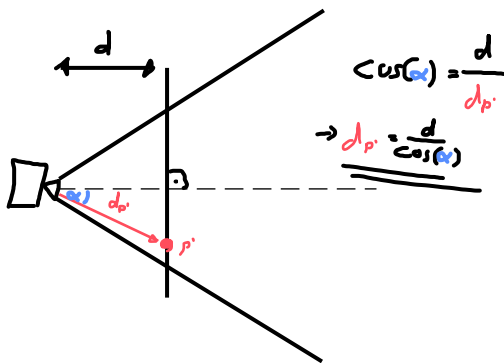


$$k \cdot \vec{P} = \vec{P}'$$

$$\rightarrow k \cdot |\vec{P}| = |\vec{P}'| \quad \text{Ähnlichkeit!}$$

$$= k \left(\frac{\vec{P} \cdot \vec{c}}{|\vec{c}|} \right) = |\vec{P}| \cdot \frac{d}{|\vec{c}|}$$

$$k = \frac{d}{\vec{P} \cdot \vec{c}}$$



Grundsätzliches Vorgehen für die Projektion eines Punktes auf den Bildschirm:

Das Ziel ist es, den Punkt auf eine virtuelle Leinwand, die sich senkrecht in einer festgelegten Distanz vor der Kamera befindet, zu projizieren. Dabei soll die Perspektive beibehalten werden, das heißt, dass der Punkt um die Kamera mit einem bestimmten Faktor zentrisch gestreckt werden muss.

Erst wird der Punkt in den lokalen Koordinaten der Kamera angegeben. Das heißt, dass das ab jetzt die Position des Punktes so angegeben wird, wie es für jemanden wäre, der die Kamera hält. Zum Beispiel würde dann ein Punkt gerade aus vor der Kamera nun die Koordinaten (0, 1) haben, egal wo er sich im globalen Koordinatensystem befindet, oder wie die Kamera um ihn gedreht ist.

Dann wird der Punkt so um die Kamera zentrisch gestreckt, dass er auf der virtuellen Leinwand landet. Da die Leinwand ja immer konstant vor der Kamera bleibt und sich diese im lokalen Kamera-Koordinatensystem weder bewegt noch dreht, ist diese Leinwand immer parallel zum Bildschirm und wir können nun einfach die z-Komponente der projizierten Punkte weglassen, da die ja immer einfach die z-Koordinate der Leinwand ist, und erhalten so die Bildschirm Koordinaten des Projizierten Punktes. Jetzt müssen wir sie nur noch so skalieren, dass die Koordinaten der Ecken der Leinwand mit den Koordinaten der Ecken des Bildschirms übereinstimmen und wir können nun Punkte Projizieren.

Methode um die Kamera um jede Achse drehen zu können:

Das Problem ist, dass wir die Kamera nicht einfach um die zB x-Achse der Kamera drehen können, wenn diese x-Achse nicht mit der globalen x-Achse übereinstimmt. Eine Idee, um das dennoch hinzukriegen ist es, die Kamera erst so zu drehen, dass ihre Achsen mit den Achsen des globalen Koordinatensystems wieder übereinstimmen und uns diese Winkel zu merken. Dann wenden wir die gewünschten Rotationen der Kamera einfach jetzt an, und drehen die Kamera mit den negativen Winkeln von vorher zurück. Intuitiv sollte das die Kamera insgesamt wie gewollt gedreht haben.