Pixel values: 3, 13, 50, 195, 255

Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)

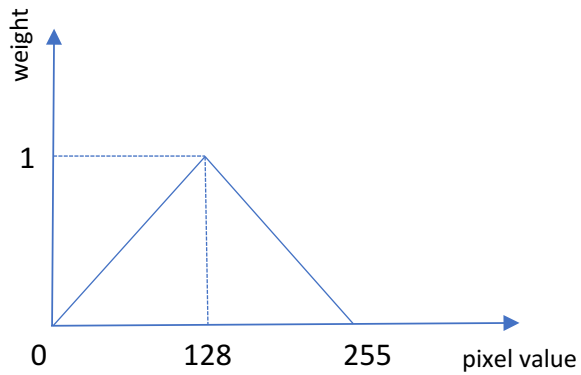
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



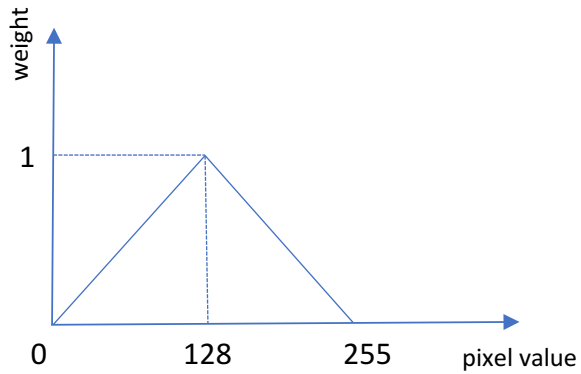
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



w1= 3/128= 0.0234

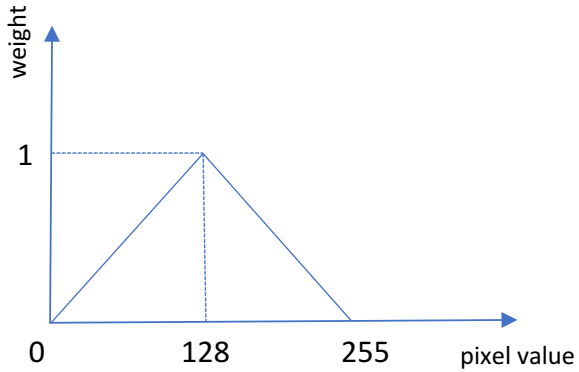
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



w1= 3/128= 0.0234
w2 = 13/128 = 0.1015
w3 = 50/128= 0.3906

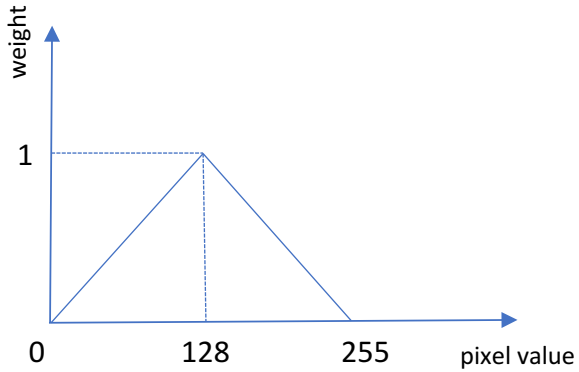
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



w1= 3/128= 0.0234
w2 = 13/128 = 0.1015
w3 = 50/128= 0.3906
w4 = (255 - 195)/128 = 0.46875

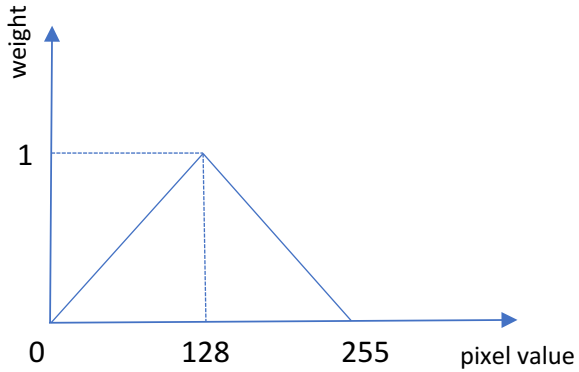
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



w1= 3/128= 0.0234
w2 = 13/128 = 0.1015
w3 = 50/128= 0.3906
w4 = (255 - 195)/128 = 0.46875
wn = w1 + w2 + w3 + w4 = 0.98435

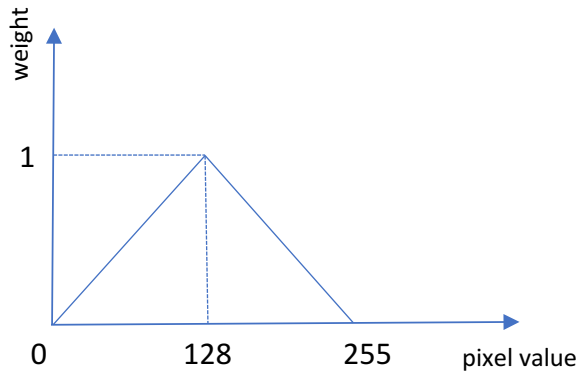
Pixel values: 3, 13, 50, 195, 255

Step 1: Remove over- and under-exposed pixel values
3, 13, 50, 195, ~~255~~

Step 2: Map to [0,1] range
3/255, 13/255, 50/255, 195/255

Step 3: Compute exposure factors (2 stops apart)
3/255, (13/255)/4, (50/255)/16, (195/255)/64

Step 4: Compute weights for the following expression
3/255*(w1/wn) + ((13/255)/4)*(w2/wn) + ((50/255)/16)*(w3/wn) + ((195/255)/64)*(w4/wn)



w1= 3/128= 0.0234
w2 = 13/128 = 0.1015
w3 = 50/128= 0.3906
w4 = (255 - 195)/128 = 0.46875
wn = w1 + w2 + w3 + w4 = 0.98435

Step 5: Substitute values into the formula
HDR = 0.00027967+ 0.001314 + 0.00486305 + 0.0056899 = 0.01214706.

Pixel values: 0, 0, 12, 45, 182

Pixel values: 0, 0, 12, 45, 182

Step 1: Remove over- and under-exposed pixel values

~~0~~, ~~0~~, 12, 45, 182

Pixel values: 0, 0, 12, 45, 182

Step 1: Remove over- and under-exposed pixel values
~~0~~, ~~0~~, 12, 45, 182

Step 2: Map to [0,1] range
12/255, 45/255, 182/255

Pixel values: 0, 0, 12, 45, 182

Step 1: Remove over- and under-exposed pixel values
~~0~~, ~~0~~, 12, 45, 182

Step 2: Map to [0,1] range
12/255, 45/255, 182/255

Step 3: Compute exposure factors (2 stops apart)
(12/255)/16, (45/255)/64, (182/255)/256

Pixel values: 0, 0, 12, 45, 182

Step 1: Remove over- and under-exposed pixel values
~~0~~, ~~0~~, 12, 45, 182

Step 2: Map to [0,1] range
12/255, 45/255, 182/255

Step 3: Compute exposure factors (2 stops apart)
(12/255)/16, (45/255)/64, (182/255)/256

Step 4: Compute weights for the uniform weighting function
3 pixel values -> each weight = 1/3

Pixel values: 0, 0, 12, 45, 182

Step 1: Remove over- and under-exposed pixel values
~~0~~, ~~0~~, 12, 45, 182

Step 2: Map to [0,1] range
12/255, 45/255, 182/255

Step 3: Compute exposure factors (2 stops apart)
(12/255)/16, (45/255)/64, (182/255)/256

Step 4: Compute weights for the uniform weighting function
3 pixel values -> each weight = 1/3

Step 5: Substitute values into the formula
HDR = ((12/255)/16)*1/3 + ((45/255)/64)*1/3 + ((182/255)/256)*1/3 =
= 0.000980392 + 0.000919117 + 0.00092933 = 0.0028288

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

-> scale down by factor 4.0

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

-> scale down by factor 4.0

0.1/4 = 0.025

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

-> scale down by factor 4.0

0.1/4 = 0.025

iii) exposure = 0.2 sec, aperture = f/8, ISO = 400, ND = 1.0

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

-> scale down by factor 4.0

0.1/4 = 0.025

iii) exposure = 0.2 sec, aperture = f/8, ISO = 400, ND = 1.0

scale up by factor 4.0 (for ISO) and down by factor 10.0 (for ND)

Camera settings: exposure = 0.2 sec, aperture = f/8, & ISO = 100
Irradiance value: 0.1

i) exposure = 0.4 sec, aperture = f/8, ISO = 100

double exposure -> scale up by factor 2.0

0.1*2 = 0.2

ii) exposure = 0.2 sec, aperture = f/16, ISO = 100

decreasing radius of aperture by factor of 2 means decreasing the area by factor of 4

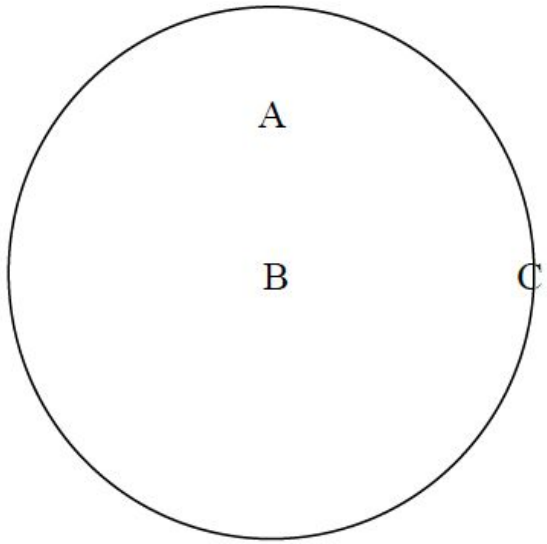
-> scale down by factor 4.0

0.1/4 = 0.025

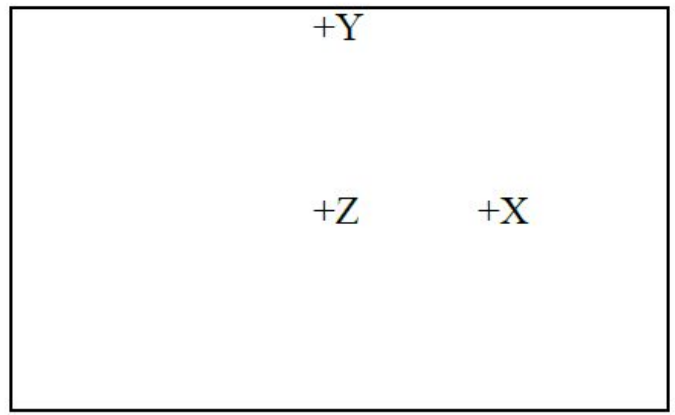
iii) exposure = 0.2 sec, aperture = f/8, ISO = 400, ND = 1.0

scale up by factor 4.0 (for ISO) and down by factor 10.0 (for ND)

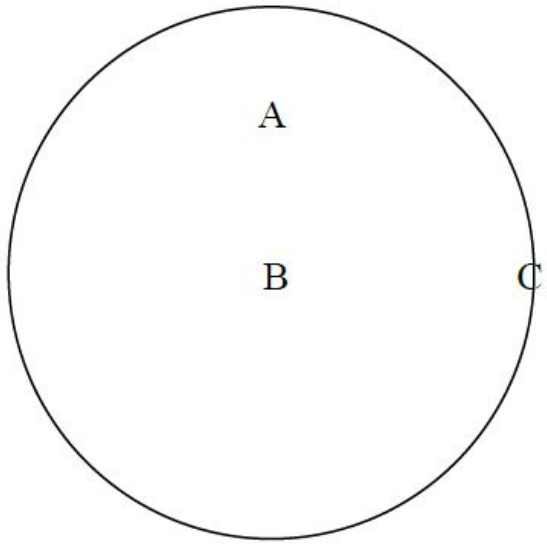
0.1*4/10 = 0.04



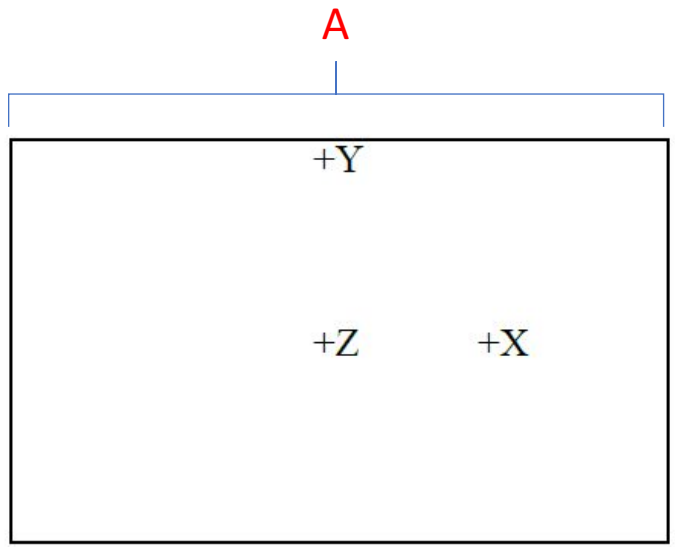
M



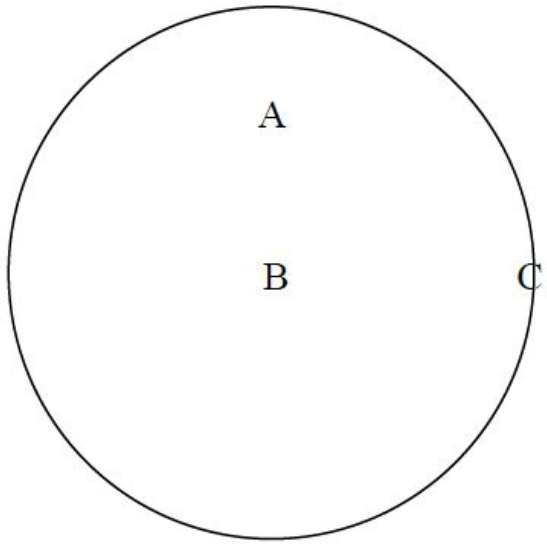
L



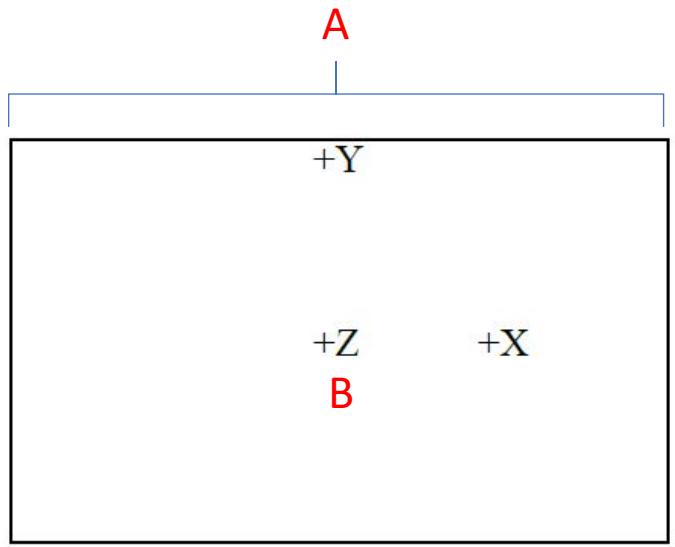
M



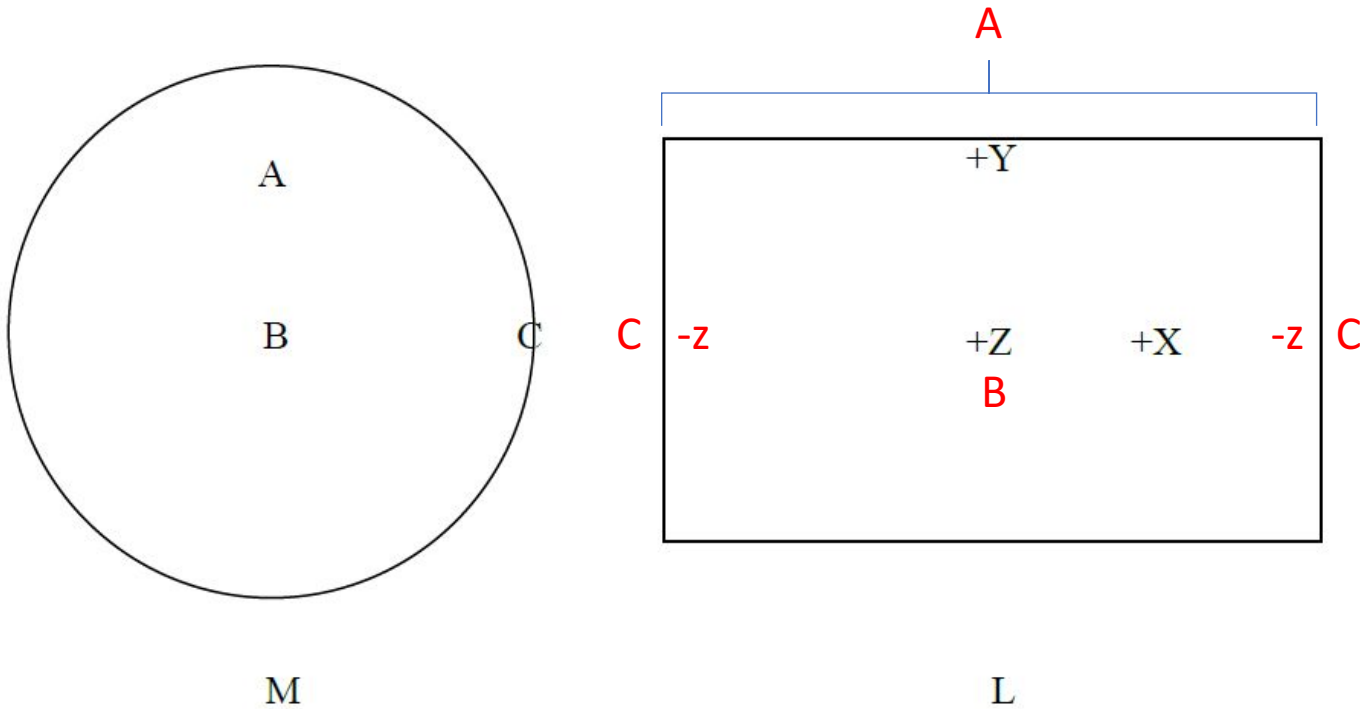
L

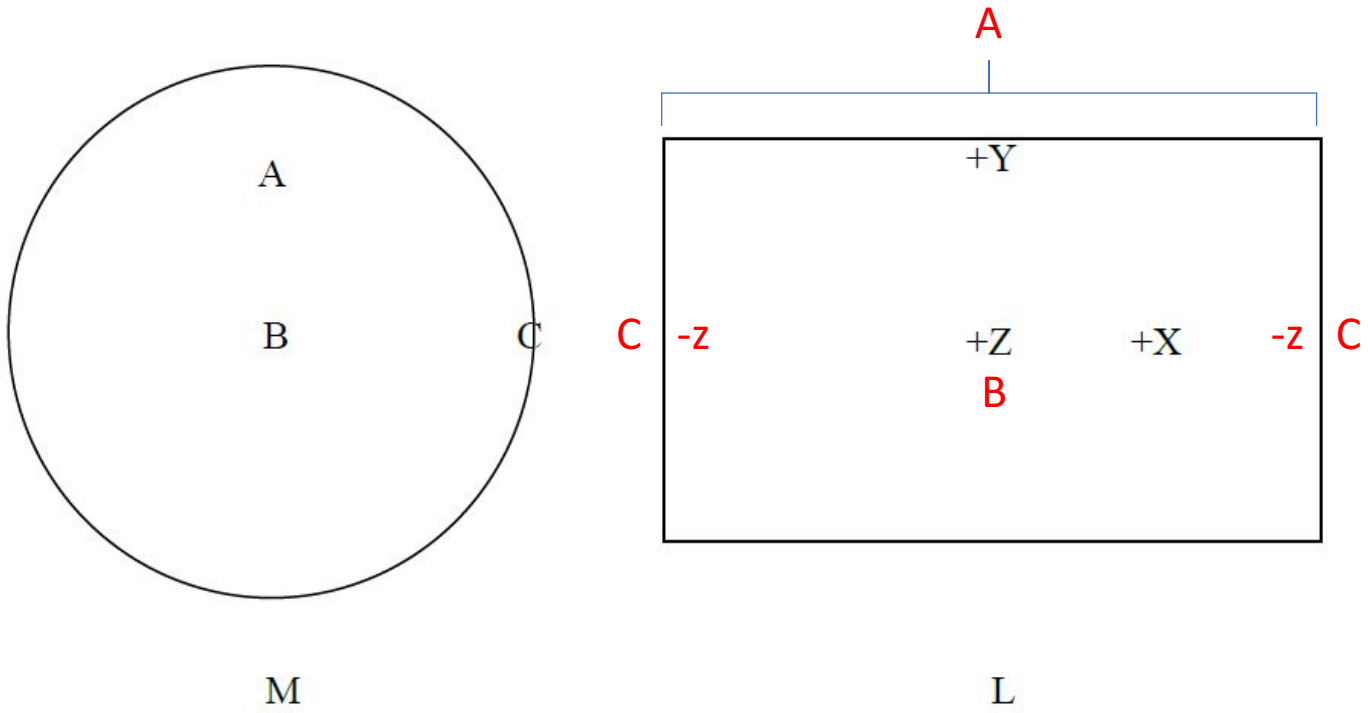


M



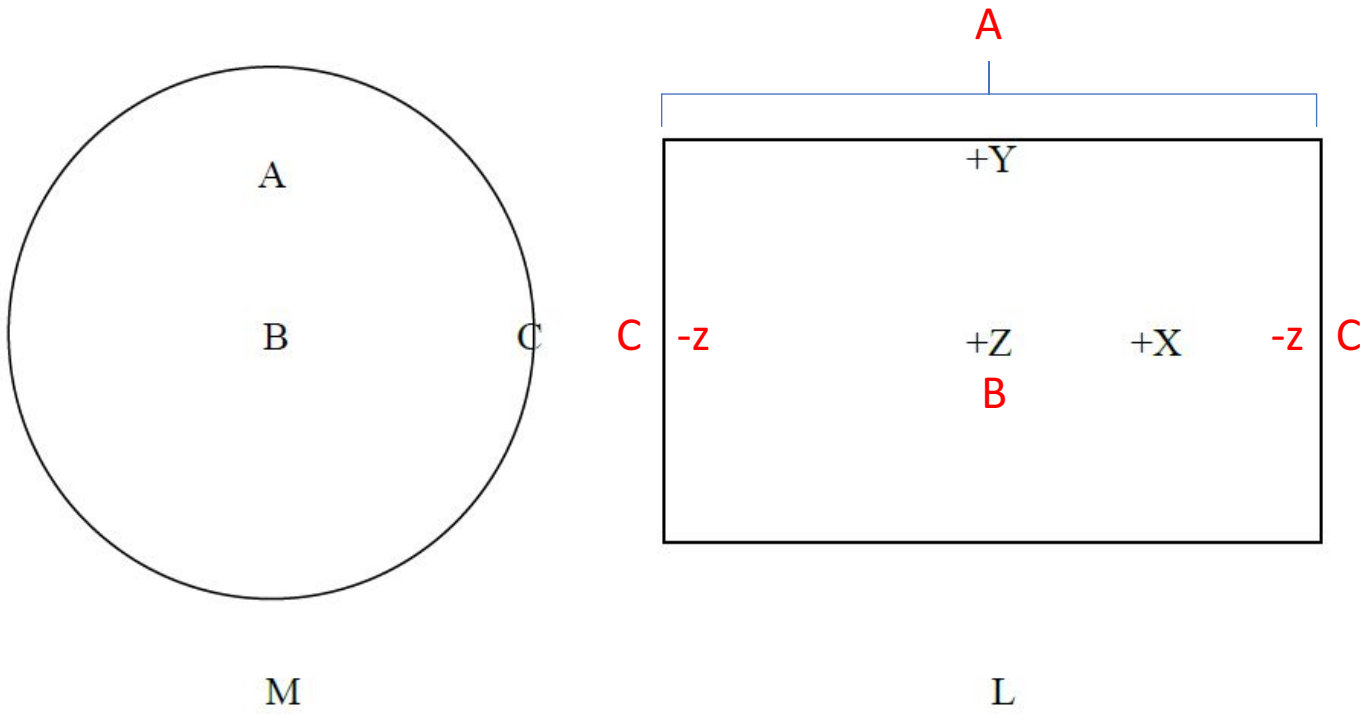
L


A
B
C
M
A
+Y
C
-z
+Z
+X
-z
C
B
L



Cartesian coordinates of A, B & C in L:

A = +Y in L = (0, 1, 0)
B = +Z in L = (0, 0, 1)
C = -Z in L = (0, 0, -1)



Cartesian coordinates of A, B & C in L:

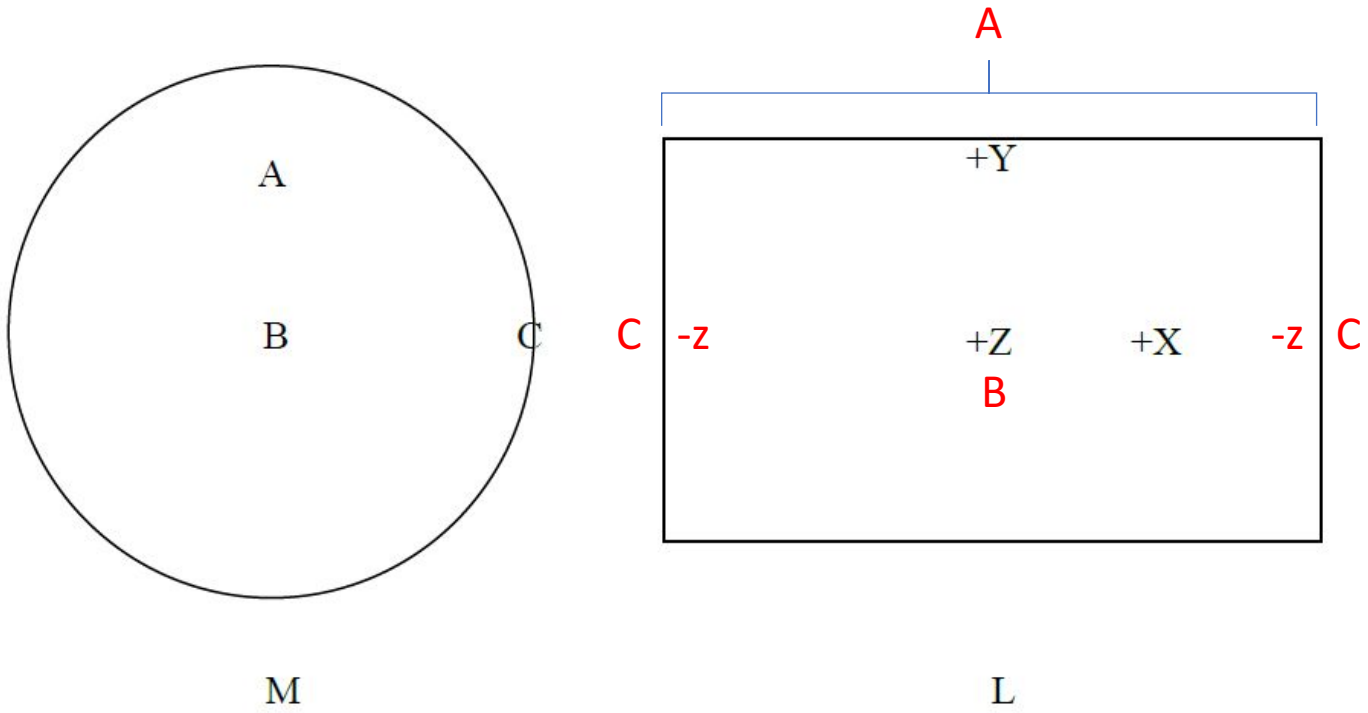
A = +Y in L = (0, 1, 0)
B = +Z in L = (0, 0, 1)
C = -Z in L = (0, 0, -1)

The surface normals of the points in M:

n_A =

n_B =

n_C =

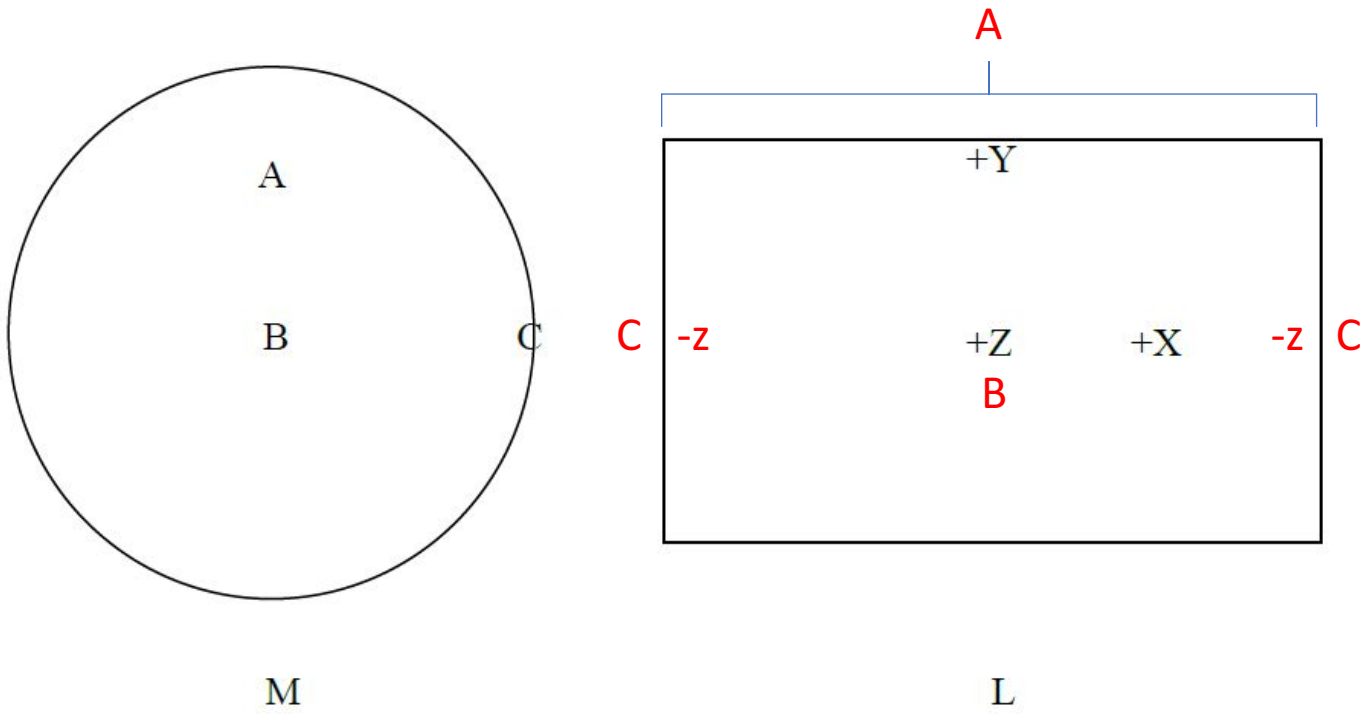


Cartesian coordinates of A, B & C in L:

A = +Y in L = (0, 1, 0)
B = +Z in L = (0, 0, 1)
C = -Z in L = (0, 0, -1)

The surface normals of the points in M:

n_A =
n_B = (0, 0, 1)
n_C =

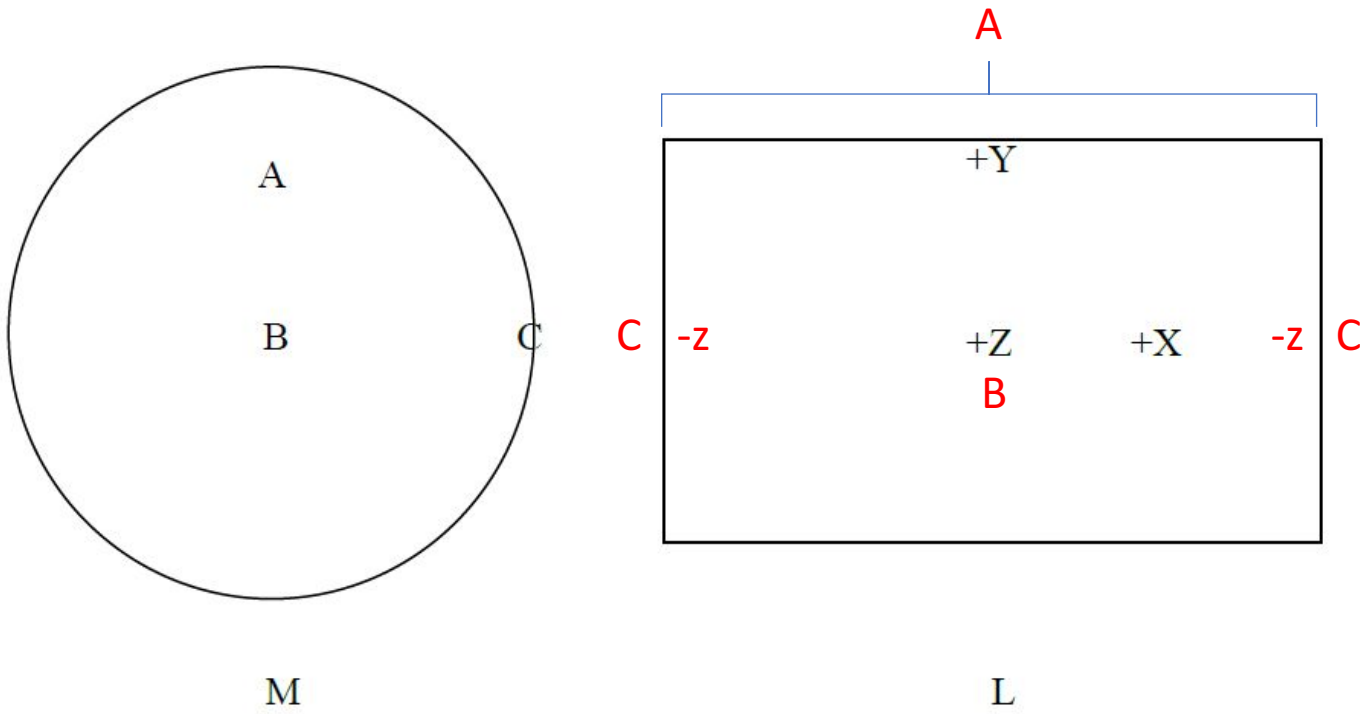


Cartesian coordinates of A, B & C in L:

A = +Y in L = (0, 1, 0)
B = +Z in L = (0, 0, 1)
C = -Z in L = (0, 0, -1)

The surface normals of the points in M:

n_A =
n_B = (0, 0, 1)
n_C = (1, 0 , 0)



Cartesian coordinates of A, B & C in L:

A = +Y in L = (0, 1, 0)
B = +Z in L = (0, 0, 1)
C = -Z in L = (0, 0, -1)

The surface normals of the points in M:
$n_A = (0, 1/\sqrt{2}, 1/\sqrt{2})$
$n_B = (0, 0, 1)$
$n_C = (1, 0 , 0)$

Obtained using the half vector equation:
$\mathbf{n} = (\mathbf{v} + \mathbf{l})/||\mathbf{v} + \mathbf{l}||$