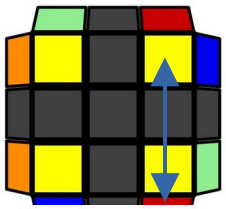
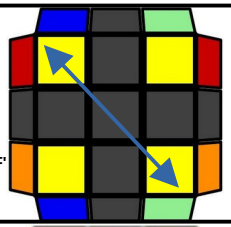
O



Adjacent Swap (inv)

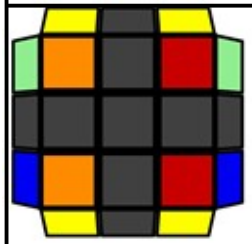
R U R' F' R U R' U' R' F R2 U' R'

Diagonal Swap (inv)



F R U' R' U' R U R' F' R U R' U' R' F R F'

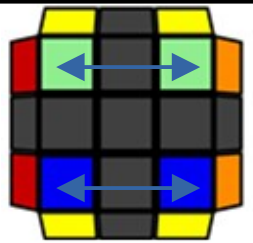
H



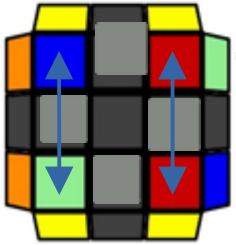
Columns (inv)

R U2 R' U' R U R' U' R U' R'

Rows (inv)



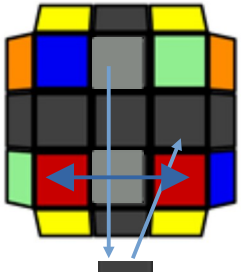
F (R U R' U') (R U R' U') (R U R' U') F'



Column (Pi X)

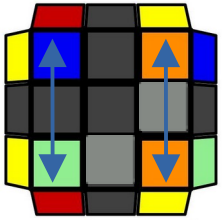
R U2' R' (R' F R F') U2 (R' F R F')

Row (Pi col.)



(U2) r U' r2' D' r U' r' D r2 U r

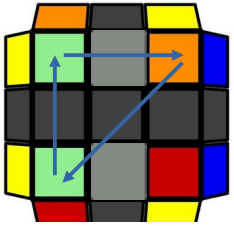
Pi



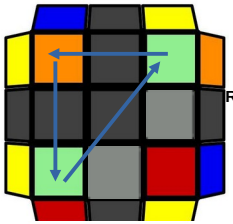
Right Bar (inv.)

F (R U R' U') (R U R' U') F'

Left Bar (inv.)



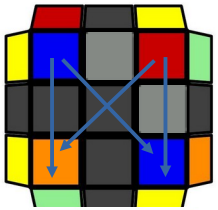
(U') R' U' (R' F R F') R U' R' U2 R



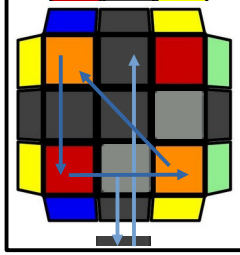
Forward Slash

R U2 R' U' R U R' U2' (R' F R F')

Back Slash



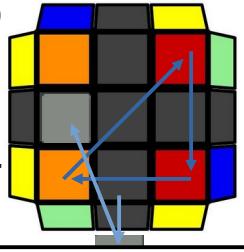
(U) (F R' F' R) U2 (R U' R' U) R U2' R'



X (H col.)

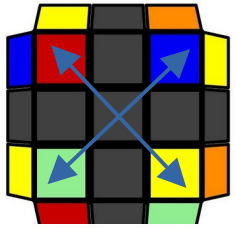
(U') R' F R U F U' (R U R' U') F'

Columns (H row)



(U') r U' r2' D' r U r' D r2 U r'

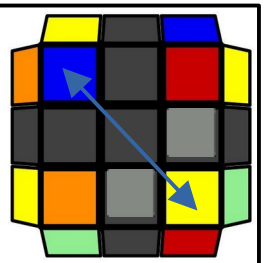
S



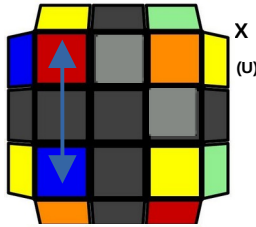
Left Bar

(U) R U R' U R U2 R'

Right Bar



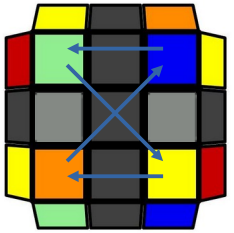
(U') R U R' U (R' F R F') R U2' R'



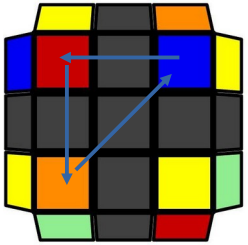
X

(U) L' U2 L U2' L F' L' F

Columns



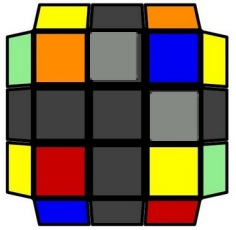
(U) (R U R' U') (R' F R F') R U (R' U R U') U'R'



Back Slash

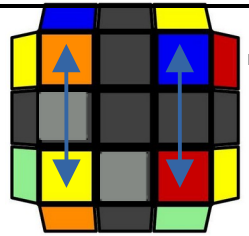
(U) R U' L' U R' U' L

Forward Slash



(U) F R' F' R U2 R U2' R'

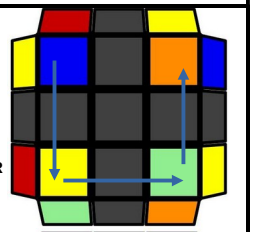
As



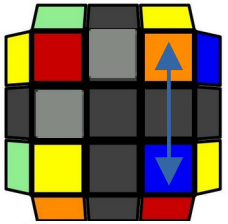
Left Bar

R' U' R U' R' U R' F R F' U R

Right Bar



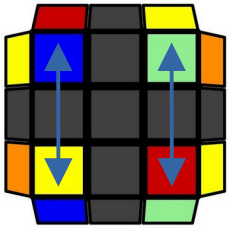
(U) R' U' R U' R' U2' R



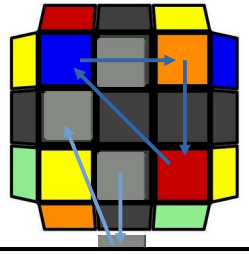
X

(U') R U2' R' U2 (R' F R F')

Columns



(U') R2 D R' U R D' (R' U R' U') R U' R'

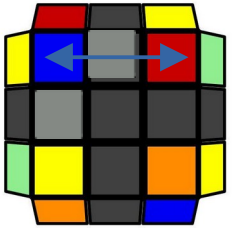


Forward Slash

(U') (R' F R F') r U r'

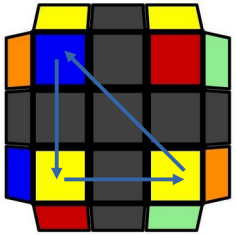
(U') L' U R U' L U R'

Back Slash



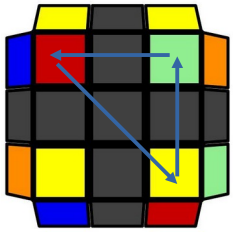
(U') F' L F L' U2' L' U2 L

U

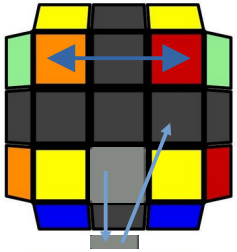


Forward Slash | back Slash

(U2) R2 D R' U2 R D' R' U2 R'



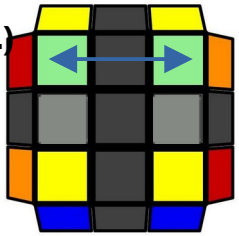
R2' D' R U2 R' D R U2 R



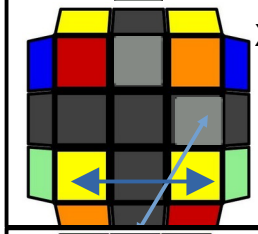
Front Row (T row)

R2' F U' F U F2 R2 U' R' F R

Rows (inv.)



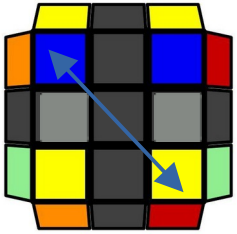
(U') F R2 D R' U R D' R2' U' F'



X (T back row)

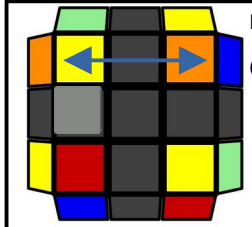
(U2) r U' r' U r' D' r U' r' D r

Back Row (inv.)



(U') F (R U R' U') F'

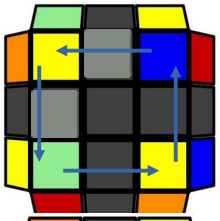
L



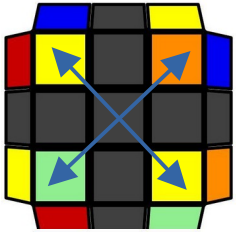
Mirror (T right bar)

(U2) F R U' R' U' R U R' F'

Inverse



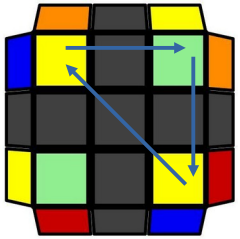
(U2) (F R' F' R) (U R U' R')



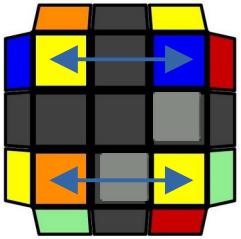
Pure

R U2 (R' U' R U) (R' U' R U) R' U' R U' R'

Front Commutator (U /)



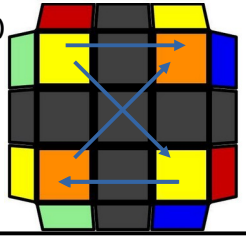
(U2) R U2 R D R' U2 R D' R2'



Diag

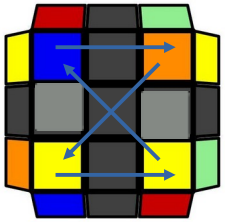
(U2) R U2' R2' F R F' R U2' R'

Back Commutator (U \)



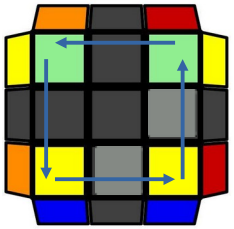
(U) R' U2 R' D' R U2 R' D R2

T



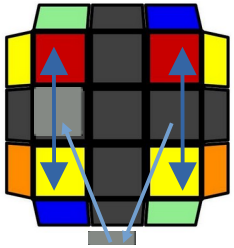
Left Bar (L inverse)

(U') (R U R' U') (R' F R F')



Rows (Ufrow^-1)

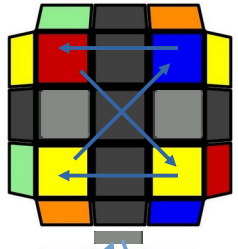
F R' F R2 U' (R' U' R U) R' F2



Back Row (U x)

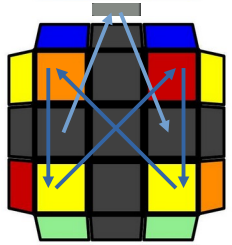
r' D' r U r' D r U' r U r'

Right Bar (L mirror)



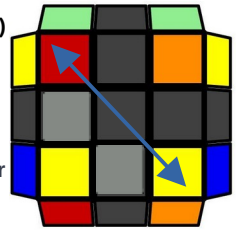
(U) (L' U' L U) (L F' L' F)

Front Row (inv.)



r' U r U2' R2' F R F' R

Columns (inv.)



(U2) r2' D' r U r' D r2 U' r' U' r

- This document is a “cheat sheet” (easy to print) for beginner with CMLL’s (Roux method) adapted from Kian’s document : https://docs.google.com/document/d/1S_y2evAxJ2vMMFDtfGSOsgHuDMYR-sEjoEU_mtSD7Qc/edit

- Grey squares indicates edges that see theire orientation flipped (The position also may be flipped, in this case if it goes to bottom side then new position will be show by light blue arrow (helps for EO)).

- In parenthesis there is the name of the opposite algs (if they are in a same line, then there is just a pipe |, and “inv.” for involutions. (the alg is his own opposite))

- Dark blue arrows can help recognizing the case / understanding moves.

- This doc isn’t finished, a lot of things to do so feel free to contribute : https://github.com/Adrien-No/cmll

- Contact me : adrien.no@proton.me

Color Coding
Green = R U R' U' Family
Blue = R U R' U R U2 R' Family
Orange = R F' R' F Family