Cell Reports Methods, Volume *3*

Supplemental information

Functional imaging-guided cell selection for evolving genetically encoded fluorescent indicators

Chang Lin, Lihao Liu, and Peng Zou

Supplementary Table 1. The yield and precision of Faculae, related to Figure 1

Experiment	Number of Activated cells	Number of recovered PAmCherry$^+$ cells	Yield	GFP$^+$/ miRFP680$^-$	GFP$^-$/ miRFP680$^+$	GFP$^-$/ miRFP680$^-$	Precision
#1	90	44	49%	40	1	3	91%
#2	50	19	38%	19	0	0	100%

Supplementary Table 2. Screening count of each library, related to Figure 2, 4 and 6

Library	Number of screened cells	Number of recovered PAmCherry$^+$ cells	Number of sequencing results	Number of templates in result
miRFP680 library	0.4×10^5	48	17	12[a]
NIR-GECO1/2G mixed library	0.6×10^5	47	14	13[b]
mIFP library	1.1×10^5	43	22	8[b]
CaM library	1.0×10^5	36	15	7[b]
Combined library	1.2×10^5	81	21	0[b]
iBB library-R1	0.4×10^5	800	-[c]	-[c]
iBB library-R2	1.9×10^5	88	21	8[d]

[a] In this case, the template represents miRFP680. [b] In these cases, the template represents NIR-GECO2G. [c] In this case, cells were collected into one tube for PCR amplification of target gene. [d] In this case, the template represents Nier1s.

Supplementary Table 3. Characterization of FR-GECI mutants in HEK 293T, related to Figure 4

Mutants	Relative brightness[a]	Relative molecular brightness (normalized against EGFP)[a]	SBR[b]	Half time ($t_{1/2}$) of photobleaching (s)[c]
VNS (NIR-GECO2G)	1.00 ± 0.03	1.00 ± 0.02	1.00 ± 0.03	125 ± 4
QKM (Nier1s)	1.48 ± 0.05	1.47 ± 0.03	1.40 ± 0.10	155 ± 10
ETC (Nier1b)	2.55 ± 0.05	3.50 ± 0.11	1.00 ± 0.03	171 ± 11
STM	1.30 ± 0.12	1.67 ± 0.06	1.51 ± 0.08	132 ± 4
DRF	1.98 ± 0.06	1.81 ± 0.04	1.45 ± 0.10	141 ± 10
GCI	1.65 ± 0.08	1.70 ± 0.03	1.27 ± 0.09	150 ± 10
GRH	2.61 ± 0.16	2.82 ± 0.11	1.16 ± 0.05	142 ± 11
QRC	1.89 ± 0.13	2.77 ± 0.14	1.12 ± 0.04	-
KIV	2.41 ± 0.07	3.15 ± 0.06	1.05 ± 0.08	123 ± 11
RNP	2.99 ± 0.05	3.74 ± 0.06	1.04 ± 0.06	130 ± 6
KLA	2.41 ± 0.02	2.78 ± 0.11	1.04 ± 0.02	123 ± 3
KMC	2.83 ± 0.11	2.97 ± 0.07	1.03 ± 0.05	127 ± 8
TAA	1.81 ± 0.03	2.25 ± 0.11	1.03 ± 0.03	-
EEY	2.01 ± 0.13	2.60 ± 0.06	1.01 ± 0.04	-
NTT	1.86 ± 0.08	2.53 ± 0.05	1.00 ± 0.03	-
KSC	2.34 ± 0.14	3.11 ± 0.05	0.99 ± 0.05	130 ± 6

SRC	2.50 ± 0.06	2.84 ± 0.14	0.90 ± 0.05	147 ± 7
SHC	1.82 ± 0.06	2.98 ± 0.03	0.97 ± 0.07	-
DQV	3.43 ± 0.07	3.13 ± 0.04	0.96 ± 0.03	134 ± 5
QGV	2.80 ± 0.16	2.97 ± 0.06	0.88 ± 0.03	-
SGC	2.87 ± 0.23	2.72 ± 0.07	0.83 ± 0.05	-
RQP	2.74 ± 0.11	3.32 ± 0.05	0.73 ± 0.02	-

[a] Data were normalized to NIR-GECO2G. Each group contains three experimental replicates. Mean ± S.E.M

[b] Data were normalized to NIR-GECO2G. Each group contains 8-10 cells. Mean ± S.E.M

[c] Each group contains 8-10 cells. Mean ± S.E.M

Supplementary Table 4. Characterization of FR-GECI mutants in neuron, related to Figure 5

Mutants	Brightness rel. EGFP[a]	Response to 1 AP (%)[b]	Response to 10 AP (%)[b]	SNR (1AP) [b]	Half-rise time (1 AP) (ms) [b]	Half-decay time (1 AP) (s) [b]
VNS (NIR-GECO2G)	1.00 ± 0.10	-2.90 ± 0.53	-16.08 ± 2.24	11.57 ± 1.79	246 ± 17	1.46 ± 0.21
QKM (Nier1s)	1.22 ± 0.17	-5.77 ± 0.90	-23.03 ± 2.88	23.58 ± 4.00	227 ± 19	1.83 ± 0.21
ETC (Nier1b)	1.93 ± 0.24	-3.04 ± 0.62	-18.31 ± 1.32	18.75 ± 5.55	211 ± 19	1.05 ± 0.12
GRH	0.70 ± 0.14	-2.62 ± 0.62	-16.32 ± 1.99	9.55 ± 0.69	256 ± 34	1.50 ± 0.29
STM	1.20 ± 0.22	-2.91 ± 0.90	-16.25 ± 3.86	10.56 ± 4.23	331 ± 104	0.72 ± 0.15
GCI	1.21 ± 0.14	-3.60 ± 1.13	-18.16 ± 3.08	17.12 ± 4.90	207 ± 17	1.71 ± 0.30
DRF	1.23 ± 0.18	-2.81 ± 0.29	-20.70 ± 1.83	14.83 ± 1.58	241 ± 7	1.58 ± 0.10
RNP	1.67 ± 0.15	-1.95 ± 0.24	-9.97 ± 0.95	10.88 ± 3.08	303 ± 118	1.38 ± 0.43

[a] Data were normalized to NIR-GECO2G. VNS: 105 cells; QKM: 59 cells; ETC: 79 cells; GRH: 46 cells; STM: 33 cells; GCI: 99 cells; DRF: 53 cells; RNP: 88 cells. Mean ± S.E.M

[b] VNS: 10 cells; QKM: 10 cells; ETC: 10 cells; GRH: 5 cells; STM: 4 cells; GCI: 5 cells; DRF: 3 cells; RNP: 3 cells. Mean ± S.E.M

Supplementary Table 5. Characterization of NIR-GECI mutants in vitro, related to Figure 5

Indicator Name	$[Ca^{2+}]$ (μM)	Ex (nm)	Em (nm)	EC ($M^{-1}\times cm^{-1}$)	QY (%)	Brightness (EC*QY)	pKa	K_d (nM)	Hill Coeff. (n)
Nier1s	0	681	697	63000	3.3	2.1	5.8	146	0.96
	39	681	691	17000	0.52	0.088	4.2		
Nier1b	0	681	698	76000	4.4	3.3	4.8	308	0.91
	39	681	693	22000	3.3	0.73	4.2		
NIR-GECO2G	0	680	697	69000	3.0	2.1	5.4	194	0.94
	39	681	691	14000	1.3	0.18	4.1		

Supplementary Table 6. Characterization of Nier1s mutants in HEK 293T, related to Figure 6

Mutants	Relative brightness[a]	SBR[b]
Nier1s	1.00 ± 0.03	1.00 ± 0.03
Y271C	0.91 ± 0.02	0.98 ± 0.03
S373Y	1.54 ± 0.07	1.15 ± 0.02
S447D	0.79 ± 0.01	1.14 ± 0.09
323 Δ GG/S447T	0.77 ± 0.02	1.14 ± 0.07
Y271F/I378T	1.24 ± 0.09	1.10 ± 0.03
E276D/Q428L	1.87 ± 0.07	0.90 ± 0.03
I378C	1.27 ± 0.09	1.01 ± 0.03
323 Δ GG/S447N	0.77 ± 0.02	1.01 ± 0.02
323 Δ GG	0.82 ± 0.04	1.06 ± 0.05

[a] Data were normalized to Nier1s. Each group contains two experimental replicates. Mean ± S.E.M

[b] Data were normalized to Nier1s. Each group contains 10 cells. Mean ± S.E.M

Supplementary Table 7. List of primers used in this study, related to STAR Methods

Primer name	Primer sequence (5'-3')
Head primer	ATGTCGGTACCGCTGACTACC
Tail primer	TTTGGACTGAGACTGTGCAAAGCTCTC
Forward primer-1	GCAACGTGCTGGTTATTGTGCTGTCTCATCATTTTG
Reverse primer-1	ACCCCTCCATGTGCACCTTGAAGCG
Forward primer-2	GGCAAAGAATTGGGGTTTGTCTGGTCAACC
Reverse primer-2	CCATGGATCCAGGGCCAGGGTTC
F60-F	GTTCCTGAACACCAACNNKGTTGTTGGCCGTCCGC
F60-R	GCGGACGGCCAACAACMNNGTTGGTGTTCAGGAAC
A125-F	GGAACCAGCAACCAAGNNKACTAACATTGCGCCGG
A125-R	CCGGCGCAATGTTAGTMNNCTTGGTTGCTGGTTCC
V133-F	CATTGCGCCGGCTCTGNNKGGTGCGCTTCATCGTA
V133-R	TACGATGAAGCGCACCMNNCAGAGCCGGCGCAATG
L136-F	GGCTCTGGTCGGTGCGNNKCATCGTATCACTTCTT
L136-R	AAGAAGTGATACGATGMNNCGCACCGACCAGAGCC
F156-F	CGAAACCGCGACTATTNNKCGTGAGATTACTGGCT
F156-R	AGCCAGTAATCTCACGMNNAATAGTCGCGGTTTCG
F162-F	CCGTGAGATTACTGGCNNKGACCGTGTGATGGTAA
F162-R	TTACCATCACACGGTCMNNGCCAGTAATCTCACGG
M168-F	CGACCGTGTGATGGTANNKCGTCTCGGCGCGCTTG
M168-R	CAAGCGCGCCGAGACGMNNTACCATCACACGGTCG
L353-F	CTCGCGTAGGCATGATNNKCTGTCCGAATGTCGTC
L353-R	GACGACATTCGGACAGMNNATCATGCCTACGCGAG
C357-F	TGATTTGCTGTCCGAANNKCGTCGTGCGGACCTGG
C357-R	CCAGGTCCGCACGACGMNNTTCGGACAGCAAATCA
Q377-F	GGCGTCTACTATTCCGNNKATCGCTCGTCGCCTGT
Q377-R	ACAGGCGACGAGCGATMNNCGGAATAGTAGACGCC
L385-F	TCGTCGCCTGTACGAANNKAACCGTGTTCGCCTGC
L385-R	GCAGGCGAACACGGTTMNNTTCGTACAGGCGACGA
E402-F	TACTCCGGTTCCGCTANNKCCGCGCATCAGCCCGC
E402-R	GCGGGCTGATGCGCGGMNNTAGCGGAACCGGAGTA
V455-F	TCTGTGGGGTCTGATCNNKTGCCACCACTACGAAC
V455-R	GTTCGTAGTGGTGGCAMNNGATCAGACCCCACAGA
Y463-F	CCACTACGAACCGCGCNNKGTTCCGTCCCACATTC
Y463-R	GAATGTGGGACGGAACMNNGCGCGGTTCGTAGTGG
S466-F	ACCGCGCTACGTTCCGNNKCACATTCGCGCTGCTG
S466-R	CAGCAGCGCGAATGTGMNNCGGAACGTAGCGCGGT

A478-F	CGAAGCGCTGGCGGAANNKTGTGCGAACCGCATCG
A478-R	CGATGCGGTTCGCACAMNNTTCCGCCAGCGCTTCG
N481-F	GGCGGAAGCCTGTGCGNNKCGCATCGCGACGCTGG
N481-R	CCAGCGTCGCGATGCGMNNCGCACAGGCTTCCGCC
A319-F	TGTACAAATGATGACANNKAAGGGTGGCGGAGGTT
A319-R	AACCTCCGCCACCCTTMNNTGTCATCATTTGTACA
A342-F	GTCACGCAGTCAGANNKATAGGTCGGCTGGG
A342-R	CCCAGCCGACCTATMNNTCTGACTGCGTGAC
D283-F	TCGCCACGTGATGACANNKCTTGGTGAGAAGTTAA
D283-R	TTAACTTCTCACCAAGMNNTGTCATCACGTGGCGA
E291-F	TGAGAAGTTAACTGATNNKGAGGTTGATGAAATGA
E291-R	TCATTTCATCAACCTCMNNATCAGTTAACTTCTCA
F235-F	CGATGGCGACGGCATCNNKGACTTCCCTGAGTTCC
F235-R	GGAACTCAGGGAAGTCMNNGATGCCGTCGCCATCG
F237-F	CGACGGCATCTTCGACNNKCCTGAGTTCCTGACGA
F237-R	TCGTCAGGAACTCAGGMNNGTCGAAGATGCCGTCG
G259-F	TGAAGAGGAAATTAGANNKGCGTTCCGCGTGTTTG
G259-R	CAAACACGCGGAACGCMNNTCTAATTTCCTCTTCA
G273-F	CGGCAATGGCTACATCNNKGCAGCAGAGCTTCGCC
G273-R	GGCGAAGCTCTGCTGCMNNGATGTAGCCATTGCCG
G347-F	AGCTATAGGTCGGCTGNNKTCGCGTAGGCATGATT
G347-R	AATCATGCCTACGCGAMNNCAGCCGACCTATAGCT
I184-F	AGAGCAGATCGCAGAGNNKAAAGAGGCTTTCTCCC
I184-R	GGGAGAAAGCCTCTTTMNNCTCTGCGATCTGCTCT
I199-F	GGACGGGGACGGGACGNNKACAACCAAGGAGCTGG
I199-R	CCAGCTCCTTGGTTGTMNNCGTCCCGTCCCCGTCC
N249-F	GATGGCAAGGAAAATGNNKGACTCAGACAGTGAAG
N249-R	CTTCACTGTCTGAGTCMNNCATTTTCCTTGCCATC
N302-F	GATCAGGGTAGCAGACNNKGATGGGGATGGTCAGG
N302-R	CCTGACCATCCCCATCMNNGTCTGCTACCCTGATC
S251-F	AAGGAAAATGAATGACNNKGACAGTGAAGAGGAAA
S251-R	TTTCCTCTTCACTGTCMNNGTCATTCATTTTCCTT
T177-F	CGCGCTTGACGATCTGNNKGAAGAGCAGATCGCAG
T177-R	CTGCGATCTGCTCTTCMNNCAGATCGTCAAGCGCG
T206-F	AACCAAGGAGCTGGGGNNKGTGTTCCGGTCTCTGG
T206-R	CCAGAGACCGGAACACCGTCCCCAGCTCCTTGGTT
T289-F	CCTTGGTGAGAAGTTANNKGATGAGGAGGTTGATG
T289-R	CATCAACCTCCTCATCMNNTAACTTCTCACCAAGG
M223-F	AGCAGAGCTGCAGGACNNKATCAATGAAGTAGATG
M223-R	CATCTACTTCATTGATMNNGTCCTGCAGCTCTGCT

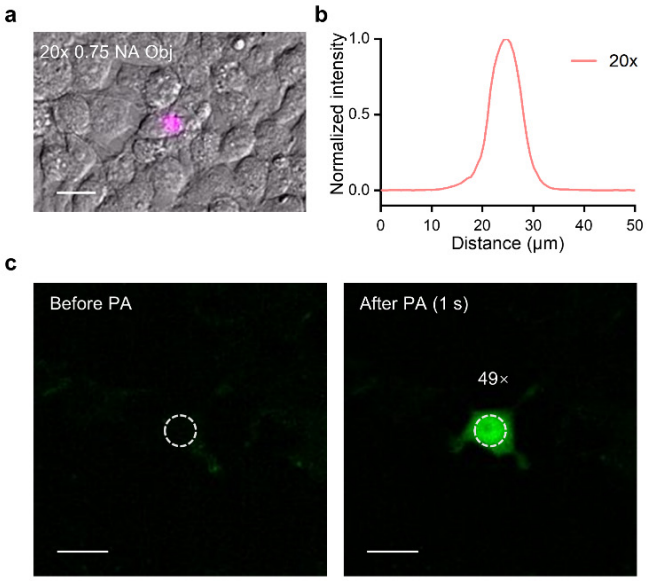
D232-F	AGTAGATGCCGATGGCNNKGGCATCTTCGACTTCC
D232-R	GGAAGTCGAAGATGCCMNNGCCATCGGCATCTACT
S251-F	AAGGAAAATGAATGACNNKGACAGTGAAGAGGAAA
S251-R	TTTCCTCTTCACTGTCMNNGTCATTCATTTTCCTT
Y271-F	TAAGGACGGCAATGGCNNKATCGGCGCAGCAGAGC
Y271-R	GCTCTGCTGCGCCGATMNNGCCATTGCCGTCCTTA
E276-F	CTACATCGGCGCAGCANNKCTTCGCCACGTGATGA
E276-R	TCATCACGTGGCGAAGMNNTGCTGCGCCGATGTAG
T282-F	GCTTCGCCACGTGATGNNKGACCTTGGTGAGAAGT
T282-R	ACTTCTCACCAAGGTCMNNCATCACGTGGCGAAGC
M316-F	CGAAGAGTTTGTACAANNKATGACAGCGAAGGGTG
M316-R	CACCCTTCGCTGTCATMNNTTGTACAAACTCTTCG
deltaGG-F	GACAGCGAAGGGTGGCTCTGTAGATTCATCAC
deltaGG-R	GTGATGAATCTACAGAGCCACCCTTCGCTGTC
GG-F	GACAGCGAAGGGTGGCGGAGGTTCTGTAGATTCATCAC
GG-R	GTGATGAATCTACAGAACCTCCGCCACCCTTCGCTGTC
S373-F	TAACCGCTACCCGGCGNNKACTATTCCGCAGATCG
S373-R	CGATCTGCGGAATAGTMNNCGCCGGGTAGCGGTTA
I378-F	GTCTACTATTCCGCAGNNKGCTCGTCGCCTGTACG
I378-R	CGTACAGGCGACGAGCMNNCTGCGGAATAGTAGAC
L391-F	TAACCGTGTTCGCCTGNNKGTAGATGTGAACTATA
L391-R	TATAGTTCACATCTACMNNCAGGCGAACACGGTTA
Y396-F	GCTGGTAGATGTGAACNNKACTCCGGTTCCGCTAG
Y396-R	CTAGCGGAACCGGAGTMNNGTTCACATCTACCAGC
Q428-F	TATGTCCCCGATCCACNNKAAATACATGCAGGACA
Q428-R	TGTCCTGCATGTATTTMNNGTGGATCGGGGACATA
S447-F	TTGCTCTCTGATGGTGNNKGGTCGTCTGTGGGGTC
S447-R	GACCCCACAGACGACCMNNCACCATCAGAGAGCAA

Supplementary Table 8. Plasmid catalog, related to STAR Methods

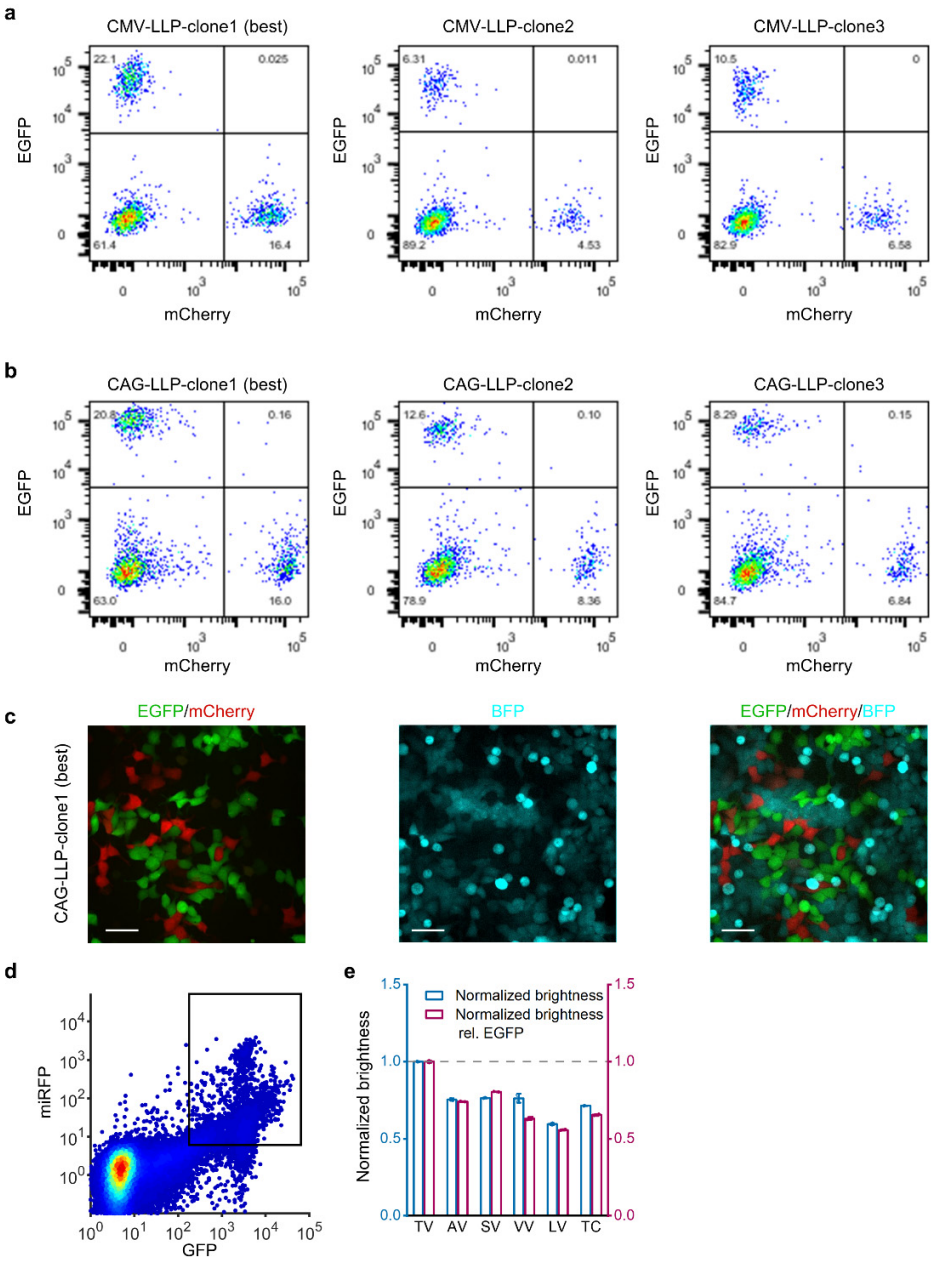
Plasmid	Description
pLX304-CAG-EBFP2-P2A-Bxb1-T2A-BSD	For CAG-LLP construction
pLX304-CMV-PAmCherry-P2A-miRFP680	Fig. 1d-f
pLX304-CMV-PAmCherry-P2A-EGFP	Fig. 1d-f
pLX304-CMV-PA-GFP	Fig. S2
attB-miRFP680-P2A-PAmCherry-NLS-EGFP-3NLS	For miRFP680 library construction
attB-EGFP	Fig. S3, S4
attB-mCherry	Fig. S3, S4
attB-NIR-GECO1 (2G)-3NLS-P2A-PAmCherry-NLS-sfGFP-3NLS	For NIR-GECO library construction. Fig. 3c-f
attB-GCaMP5g-NLS-IRES-dCherry	Fig. S7, S8
attB-GCaMP6s-NLS-IRES-mCherry	Fig. S7, S8
pcDNA3.1-NIRGECO2G-3NLS-P2A-EGFP-3NLS	For NIR-GECO characterization
pAAV-hSyn-NIR-GECO2G-IRES-EGFP-3NLS	For neural expression of NIR-GECO

Supplementary Table 9. Spectral properties and imaging apparatus for fluorescent imaging, related to STAR Methods

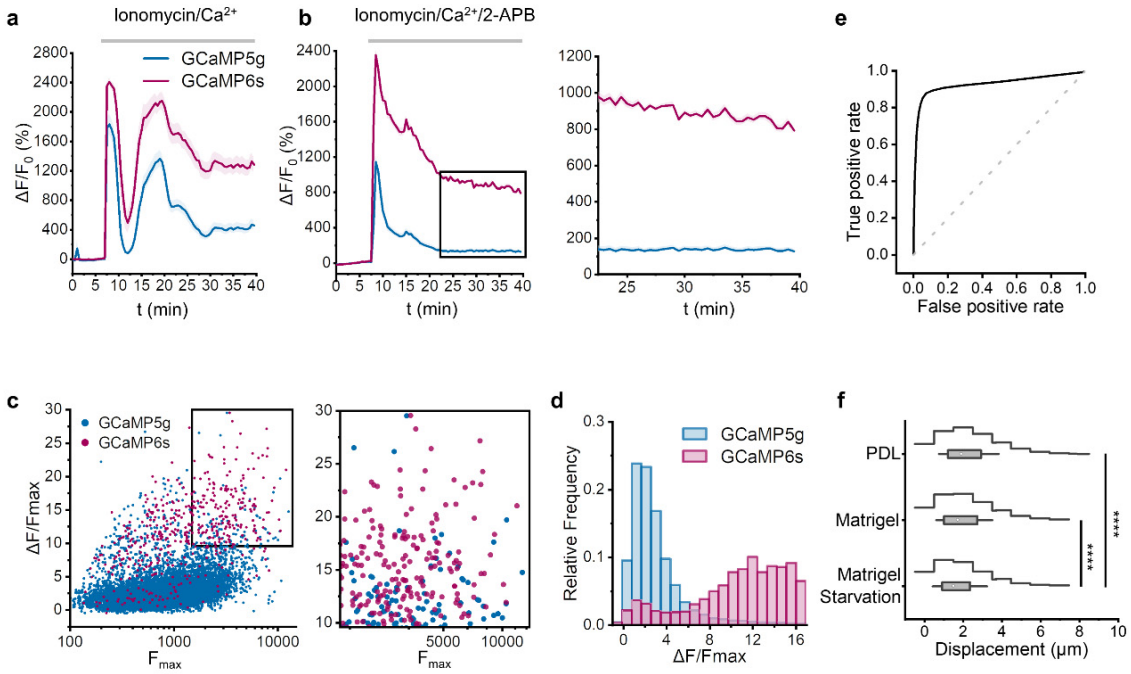
Indicator	Fluorophore	Excitation max. (nm)	Emission max. (nm)	Laser excitation wavelength (nm)	Emission filter (nm)
miRFP680	miRFP680	661	680	637	700 / 75
NIR-GECO	mIFP	683	704	637	700 / 75
EBFP2	EBFP2	383	448	405	460 / 50
mCherry	mCherry	587	610	561	630 / 75
EGFP	EGFP	489	508	488	525 / 50
GCaMP6s/GCaMP5g	cpEGFP	497	515	488	525 / 50
PAGFP	PAGFP (ON state)	504	517	488	525 / 50
PAmCherry	PAmCherry (ON state)	564	595	561	630 / 75



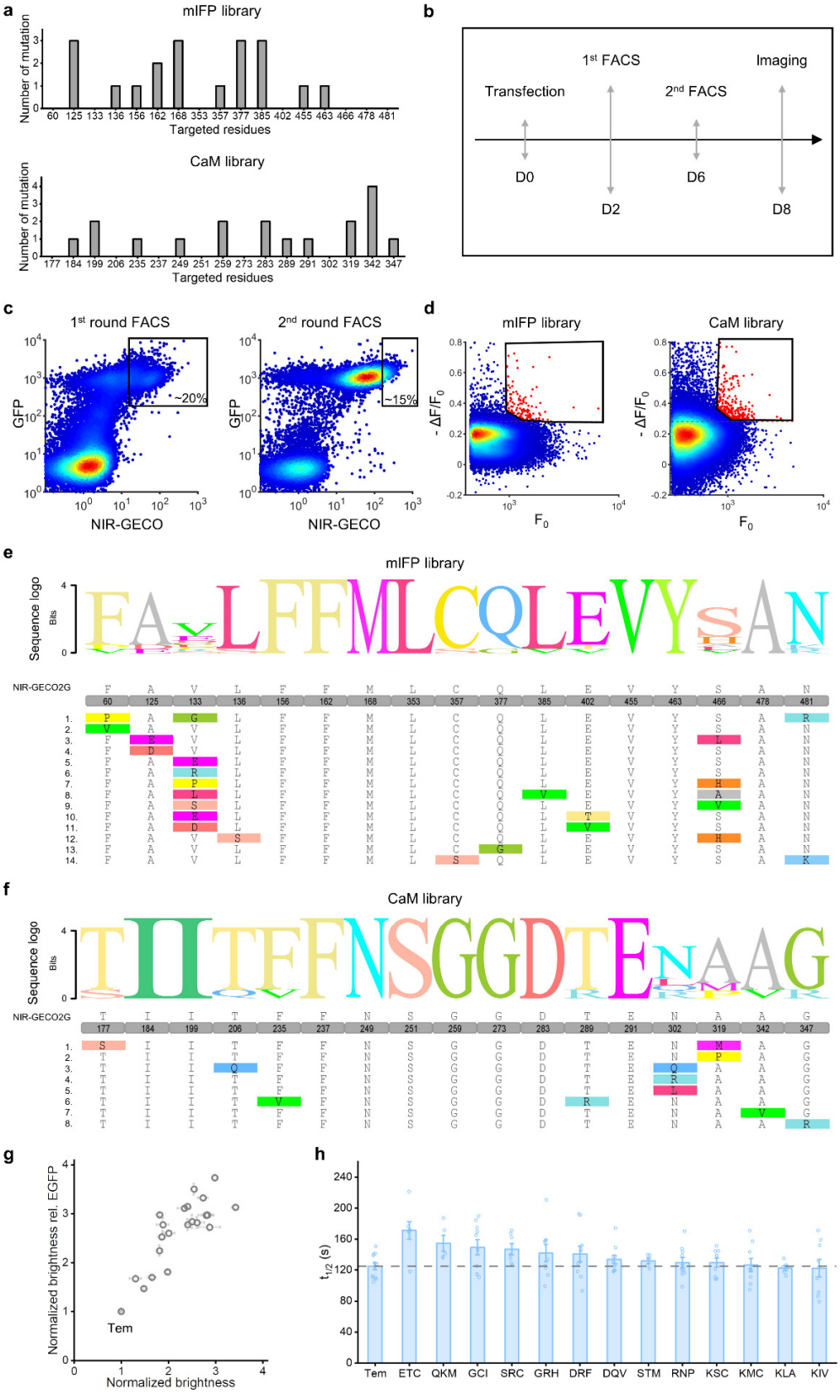
Supplementary Figure 1. Characterization of the focused spotlight for photoactivation, related to Figure 1. (A) The size of 405 nm spotlight under 20× 0.75 NA objective was determined by measuring spot size on the bottom of the glass dish. Scale bar: 20 μm. **(B)** The lateral distribution of the spot. **(C)** In cellular photoactivation of PAGFP using 405 nm focal illumination (1 s) resulted in large fluorescence change (49-fold). Dash line represents the estimated range of laser spot. Scale bar: 20 μm.



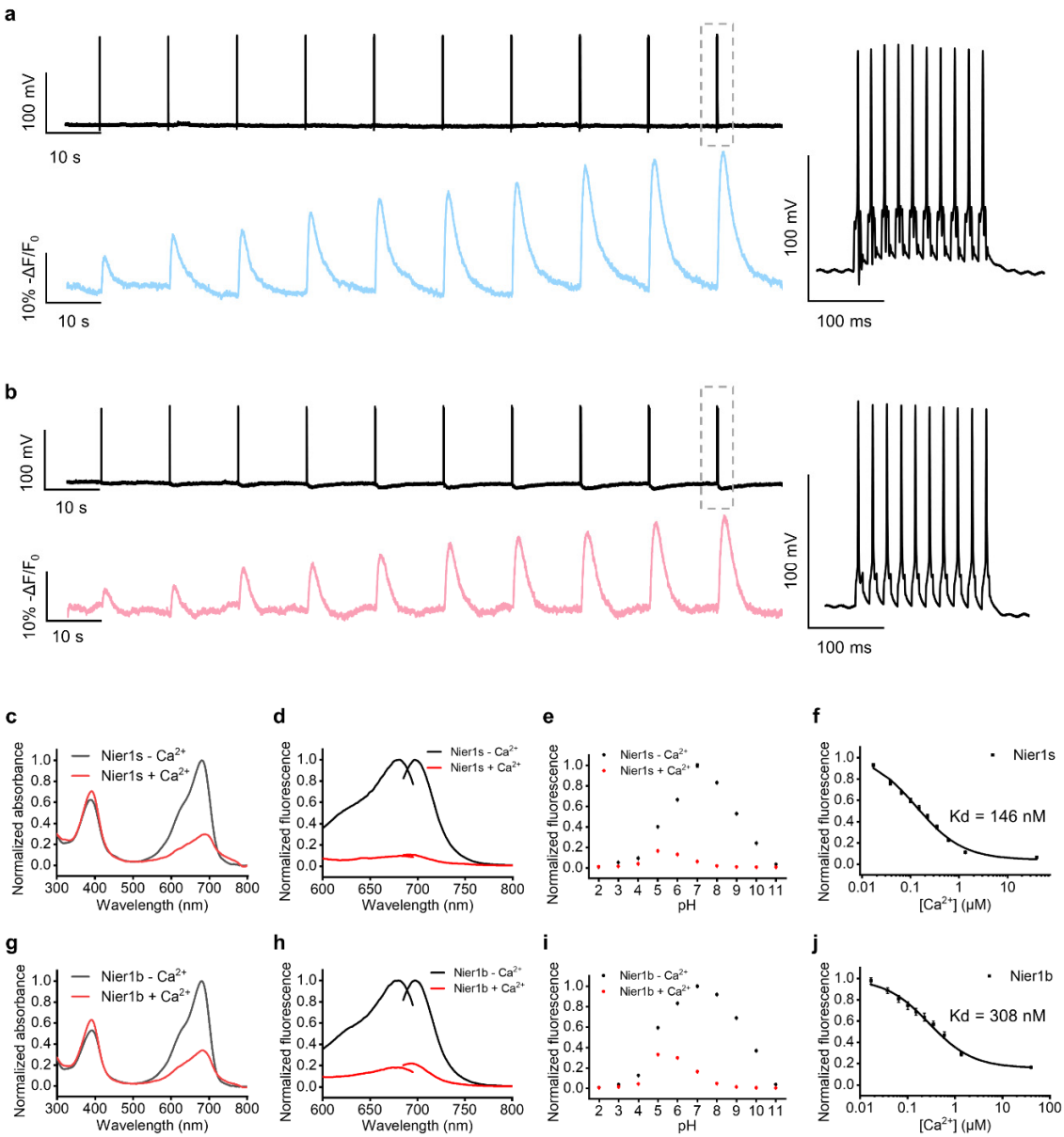
Supplementary Figure 2. Characterization of LLP cell line and miRFP680 mutants, related to Figure 2. (A-B) Characterization of LLP cell line clones. Representative cell line of CMV-LLP clones (a) and CAG-LLP clones (b) was transfected with the 1:1 plasmid mixture of attB-EGFP and attB-mCherry. The cells were analyzed by flow cytometry 3-4 days after transfection. **(C)** Fluorescent imaging of CAG-LLP-clone1 (best). In the co-transfection experiment, the recombined (BFP^-) CAG-LLP cells exclusively express either EGFP or mCherry. Scale bar: 50 μm. **(D)** FACS analysis of miRFP680 library. Two days after transfection, $miRFP680^+/GFP^+$ cells (black gate, about 5%) were collected for further imaging-based screening. **(E)** Characterization of brightness for each miRFP680 mutant in HEK 293T. The brightness is normalized relative to TV. Two replicates in each group were analyzed by flow cytometry. Error bars represent S.E.M.



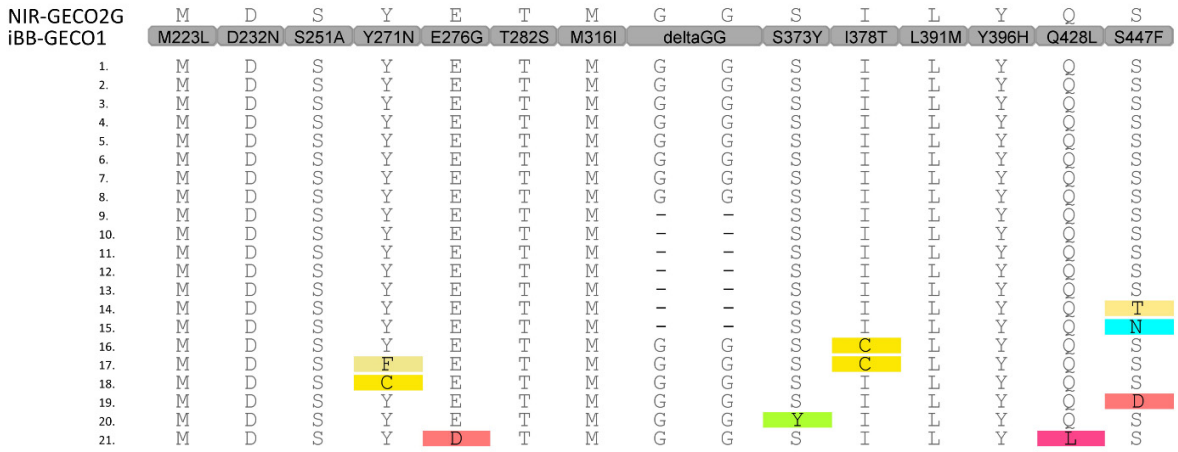
Supplementary Figure 3. Characterization of drug treatment assay, related to Figure 3. (A-B) Ionomycin/$CaCl_2$/2-APB Dosing strategy for increasing intracellular calcium level. HEK293T cells expressing either GCaMP5g-NLS-IRES-dCherry or GCaMP6s-NLS-IRES-mCherry were mixed (1:1) for time-lapse imaging. The buffer was changed 7 min after imaging started. **(A)** from EC to EC containing 2 μM Ionomycin and 1 mM $CaCl_2$. **(B)** from EC containing 50 μM 2-APB to EC containing 50 μM 2-APB, 2 μM Ionomycin and 1 mM $CaCl_2$. Right, zoomed trace from black frame in middle. 7-8 cells were counted for each group. Shaded areas represent S.E.M. **(C)** Model screening on GCaMP5g/GCaMP6s. HEK293T cells expressing either GCaMP5g-NLS-IRES-dCherry or GCaMP6s-NLS-IRES-mCherry were mixed (20:1) for two-round multi-position imaging (22×22 field of view). Brightness (F_{max}) and fluorescence change ((F_0-F_{max})/F_{max}) were calculated for each cell. Imaging buffer was changed from EC containing 50 μM 2-APB to EC containing 50 μM 2-APB, 2 μM Ionomycin and 1 mM $CaCl_2$. **(D)** The distribution of fluorescence change of GCaMP5g and GCaMP6s. **(E)** Receiver operating characteristic curve of GECI screening test. True positive rate (cell number of GCaMP6s/total cell number of GCaMP6s) and false positive rate (cell number of GCaMP6s/total cell number of GCaMP6s) were calculated at each fluorescence change from minimum to maximum. **(F)** Characterization of adherent cell migration under different treatment condition. HEK293T cells transfected with EGFP were seeded on pre-treated dishes to reach 70-80% confluence. Individual cells were tracked by imaging every 10 min. After 40 min, cell displacements were used for comparision. Data were analyzed by TrackMate (plugin in Fiji). PDL: PDL coated overnight, n_{cell} = 1173. Matrigel: Matrigel coated overnight, n_{cell} = 1538. Matrigel+Starvation: Matrigel coated overnight, serum starvation overnight, n_{cell} = 1167. **** represents $p<0.0001$. Boxes represent 25%~75% range. Error bars represent SD.


a
mIFP library
Number of mutation
Targeted residues
CaM library
b
1st FACS
Imaging
Transfection
2nd FACS
D0
D2
D6
D8
c
1st round FACS
2nd round FACS
GFP
NIR-GECO
~20%
~15%
d
mIFP library
CaM library
- ΔF/F0
F0
e
mIFP library
Sequence logo
Bits
NIR-GECO2G
f
CaM library
g
Normalized brightness rel. EGFP
Normalized brightness
Tem
h
t1/2 (s)
Tem
ETC
QKM
GCI
SRC
GRH
DRF
DQV
STM
RNP
KSC
KMC
KLA
KIV

Supplementary Figure 4. Characterization of mIFP library, CaM library and Combined library, related to Figure 4. (A) The distribution of mutation sites within mIFP library and CaM library of NIR-GECO2G. In each library, the distribution of mutations on selected sites was counted from 10 Sanger results. **(B)** Timeline of image-based pooled screening for FR-GECIs. **(C)** FACS analysis of mIFP library of NIR-GECO2G. The mIFP cell library was enriched by FACS twice. Black gates represent enriched cells. **(D)** Imaging-based screening result of mIFP library and CaM library of NIR-GECO2G. Black gates represent the range of selected cells (red scatter) for photoactivation. **(E-F)** Sequencing result and the sequence logo of screening result of mIFP **(E)** library and CaM **(F)** library from each recovered single cell. Sequencing results identical to the template were not included in the statistics. **(G)** HEK293T cells expressing NIR-GECO-mut-NLS-P2A-EGFP-NLS were analyzed by flow cytometry. The mean brightness value of each group was calculated from three biological replicates. Each mutant's brightness was normalized to template (NIR-GECO2G). **(H)** Photobleaching test of each mutant. HEK293T cells expressing NIR-GECO-mut-NLS-P2A-EGFP-NLS were illuminated by 637 nm at 9.0 W/cm^2. n_{cell} = 5-10 for each mutant. Error bars represent S.E.M.



Supplementary Figure 5. Characterization of Nier1s and Nier1b in neurons and *in vitro*, related to Figure 5. (A-B) Representative electrophysiological traces and fluorescence traces of Nier1s-expressing neuron **(A)** or Nier1b-expressing neuron **(B)** in response to current injection (1 to 10 AP). The fluorescence traces were photobleaching calibrated. **(C-J)** Absorbance spectra of Nier1s **(C)** and Nier1b **(G)** in the presence (39 μM) and absence of Ca^{2+}. Fluorescence excitation and emission spectra of Nier1s **(D)** and Nier1b **(H)** in the presence (39 μM) and absence of Ca^{2+}. pH titration curves of Nier1s **(E)** and Nier1b **(I)** in the presence (39 μM) and absence of Ca^{2+}. $\Delta F/F_0$ of Nier1s **(F)** and Nier1b **(J)** as a function of Ca^{2+} concentration (mean ± SD, n = 2).



Supplementary Figure 6. Screening result of iBB library R2, related to Figure 6. Sequencing result of screening result of iBB library from each recovered single cell.

Method S1. MATLAB scripts for image-based FP/sensor screening, related to STAR Methods.

```
% this program is for library screens with stardist/cellpose process(dual channels), a typical experiment is
% selecting NIR-GECO mutants with higher brightness and sensitivity in 293T cells
% (LLP cell line express NIRGECO-3xNLS-P2A-PAmCherry-NLS-EGFP-3xNLS).
% For screens that need two channels, which
% including one marker channel(SDC-GFP in this case) and one library
% channel (SDC-Cy5 in this case), please set the marker channel as the
% first channel in MDA list while acquiring, otherwise this program will
% use wrong channel for calculating.
clear all;clc;
% [WARNING] the varient 'background' should be assigned as zero if flat field
% correction is proceed.

% This script is written by Chang Lin
%% MM FIJI Startup
simulate = 0;      % 0 for screening in A317, 1 for simulation on PC.
para_sensi = -1;    % -1 for negative going sensor, 1 for positive going sensor or unknown sensor
EdgeArea = 50;      % minimum area size (um2) 200 or 300 for cell membrane, 50 for nuclei
EdgeDis = 8;        % minimum edge distance (um) 10 for cell membrane, 8 for nuclei
EXP = 1;        % 1 for 200 ms exposure, 2 for 500 ms exposure time.
tem_sensi = 0.28;      % absolute value
if para_sensi == -1
      sensi_txt = '-';
else
      sensi_txt = ' ';
end
import org.micromanager.internal.MMStudio;
import mmcorej.*;
import org.micromanager.api.*;
import ij.*;
gui = MMStudio(false);    % start up MM2.0 gui within MATLAB
mmc = gui.getCore;    % get the MM2.0 CMMCore
acq = gui.getAcquisitionEngine; % org.micromanager.acquisition.internal.AcquisitionWrapperEngine
MDA = gui.getAcquisitionManager();
IJ = ij.IJ;
slm = gui.live();      % org.micromanager.internal.SnapLiveManager
Miji;      % start up FIJI within MATLAB
cd('C:\Users\ZouOptics\Desktop\MM2.0\code\matlab');
mmc.setConfig("System","Startup");
mmc.setConfig("System","Startup");    % this line repeat is necessary for 561/594 startup
```

```
%% path selection
dir_flat = "C:\Users\ZouOptics\Desktop\MM2.0\flatfield\20230425";       % get direction of flat field on Z1
if simulate == 1
    dir_flat = "E:\pku\research_group\topic_screening_platform\flatfield\20220407";       % get direction of
flat field on PC
end
Channel_userdif = str2num(cell2mat(inputdlg({'Channel Num'},'input channel number',[1 35])))
switch(Channel_userdif)
    case{2}
        [file_flatMarker,path_flat] = uigetfile(dir_flat + "\*.*",'select bg-substrated Marker Channel flat field
image');
        file_flatMarker = string(file_flatMarker);
    case{1}
        file_flatMarker = [];
end
[file_flatLib,path_flat] = uigetfile(dir_flat + "\*.*",'select bg-substrated Library Channel flat field image');
file_flatLib = string(file_flatLib);
file_flat = [file_flatMarker file_flatLib];
[file_background1,path_flat] = uigetfile(dir_flat + "\*.*",'select marker channel black image');
[file_background2,path_flat] = uigetfile(dir_flat + "\*.*",'select library channel black image');
file_background = [string(file_background1) string(file_background2)];
% [file_ilastikModel,path_ilastikModel] = uigetfile(dir_ilastikModel + "\*.*",'select ilastik model');
path_flat = replace(path_flat,'\','/'); % for FIJI reading

%% spot ROI test and setting
% after acquisition, set up optical path and find the spot location,
% select the spot and run this section
IJ.runMacro("roiManager('reset');");
IJ.runMacro('roiManager("Add");');
MIJ.run("Set Measurements...", "area mean centroid stack redirect=None decimal=3");
IJ.runMacro("roiManager('Associate', 'true');");
IJ.runMacro('roiManager("Measure");');
spotROI = MIJ.getResultsTable();
IJ.runMacro("IJ.deleteRows(0,1);");
Xspot = spotROI(1,3);
Yspot = spotROI(1,4);
%% after configure, copy the dir_process to below
dir_process = "D:\LC_DATA\20230324_lin_CA31R2"; % get direction to process
dir = replace(dir_process,'\','/');
%% saving spot ROI
IJ.runMacro("roiManager('Select', 0);");
```

```
IJ.runMacro("roiManager('rename', 'laser spot');");
IJ.runMacro("roiManager('save selected', '" + dir +"/laser spot.zip"+ "');");
IJ.runMacro("roiManager('Deselect');");
IJ.runMacro("roiManager('Delete');");
clear spotROI;
%% please create grid and start the first round of screening
% for cellpose screening, use 20x20 or 22x22 grid
% for stardist screening, use 22x22 or 23x23 grid
% keep the MDA window open!!!
path_rawpre = dir_process + "\rawdata_pre";
mkdir(path_rawpre);
Channel_snap = strings(Channel_userdif,1);
for i = 1:Channel_userdif
list = {'SDC-LED','SDC-GFP','SDC-mCherry','SDC-Cy5'};
[indx,tf] = listdlg('PromptString',['Select channel_' num2str(i) ':'],...
                                    'SelectionMode','single',...
                                    'ListString',list);
Channel_snap(i) = list(indx);
end
pl_snap = gui.getPositionList(); % class org.micromanager.PositionList
Num_posi = pl_snap.getNumberOfPositions();
Num_grid = sqrt(Num_posi);
msp_snap = pl_snap.getPositions();
position_slice = zeros(Num_posi,3);
position_label = string(zeros(Num_posi,1));
XYStage_Label = msp_snap(1).getDefaultXYStage();
PFSStage_Label = msp_snap(1).getDefaultZStage();
PhysicalSize = double(mmc.getPixelSizeUm());
pl_snap.save(strcat(dir,'/screens.pos'));
for i = 1:Num_posi
    position_slice(i,1) = msp_snap(i).getX;
    position_slice(i,2) = msp_snap(i).getY;
    position_slice(i,3) = msp_snap(i).getZ;
    position_label(i,1) = msp_snap(i).getLabel();
end
for i = 1:Num_grid:Num_posi
    pl_snap.setPositions(msp_snap(i:i+Num_grid-1));
    if EXP == 1
        for j = 1:Channel_userdif

MDA.loadAcquisition('D:\Softwares\MM2.0\acqsetting\AcqSettings_'+Channel_snap(j)+'_multiP.txt');
```

```
            MDA.runAcquisition(Channel_snap(j),path_rawpre);
            IJ.runMacro('close();');
        end
    else
        for j = 1:Channel_userdif

MDA.loadAcquisition('D:\Softwares\MM2.0\acqsetting\AcqSettings_'+Channel_snap(j)+'_multiP_500.txt');
            MDA.runAcquisition(Channel_snap(j),path_rawpre);
            IJ.runMacro('close();');
        end
    end
end
%% start the second round of screening
% after GECO screening(iono/Ca/2-APB), add drug and wait for 10-12 min before start the second round
of screening
% after GRAB screening(DA or AEA), add drug and wait for 7 min before start the second round of
screening
% after GEVI screening(GA), add drug and wait for ? min before start the second round of screening
path_rawpost = dir_process + "\rawdata_post";
mkdir(path_rawpost);
for i = 1:Num_grid:Num_posi
    pl_snap.setPositions(msp_snap(i:i+Num_grid-1));
    if EXP ==1
        for j = 1:Channel_userdif

MDA.loadAcquisition('D:\Softwares\MM2.0\acqsetting\AcqSettings_'+Channel_snap(j)+'_multiP.txt');
            MDA.runAcquisition(Channel_snap(j),path_rawpost);
            IJ.runMacro('close();');
        end
    else
        for j = 1:Channel_userdif

MDA.loadAcquisition('D:\Softwares\MM2.0\acqsetting\AcqSettings_'+Channel_snap(j)+'_multiP_500.txt');
            MDA.runAcquisition(Channel_snap(j),path_rawpost);
            IJ.runMacro('close();');
        end
    end
end
%% start the third round of screening if it is model screening
% mCherry/dCherry or EGFP/EGFP(Y66H)
[indx,tf] = listdlg('PromptString',['Select channel_model:'],...
```

```
                                    'SelectionMode','single',...
                                    'ListString',list);
Channel_model = list(indx);
path_rawmodel = dir_process + "\rawdata_model";
mkdir(path_rawmodel);
for i = 1:Num_grid:Num_posi
    pl_snap.setPositions(msp_snap(i:i+Num_grid-1));
    if EXP ==1
        MDA.loadAcquisition(['D:\Softwares\MM2.0\acqsetting\AcqSettings_' Channel_model{1}
'_multiP.txt']);
        MDA.runAcquisition(Channel_model{1},path_rawmodel);
        IJ.runMacro('close();');
    else
        MDA.loadAcquisition(['D:\Softwares\MM2.0\acqsetting\AcqSettings_' Channel_model{1}
'_multiP_500.txt']);
        MDA.runAcquisition(Channel_model{1},path_rawmodel);
        IJ.runMacro('close();');
    end
end
[file_model,path_model,indx] = uigetfile(path_rawmodel + "\*.*",'select model image');
for i = 1:Num_grid
    file_omeopen = strcat(Channel_model{1},'_',num2str(i),file_model(end-28:end-11),num2str(i-
1,"%03d"),file_model(end-7:end));
    path_omeopen = strcat(path_rawmodel,'\',Channel_model{1},'_',num2str(i),'\');
    MIJ.run("Bio-Formats", "open=[" + strcat(path_omeopen,file_omeopen) + "] color_mode=Default
concatenate_series open_all_series split_channels view=Hyperstack stack_order=XYCZT"); % open
"before" ome-tiff file
    IJ.runMacro("rename('"+Channel_model{1}+"_"+num2str(i)+"');");
end
Conc = 'image'+string([1:Num_grid]')+'=['+Channel_model{1}+'_'+string([1:Num_grid]')+']';
MIJ.run("Concatenate...",strcat("title=",Channel_model{1}," ",join(Conc,1)));
path_save = dir + "/channel_" + Channel_model{1};
mkdir(path_save);
MIJ.run("Image Sequence... ", "format=TIFF name=stack_ save=[" + path_save + "]");
%% process multi-channel multi-time-point ome-xml tiff
[file_ome,path_ome,indx] = uigetfile(path_rawpre + "\*.*",'select pre marker channel image');
[file_ome2,path_ome2,indx] = uigetfile(path_rawpost + "\*.*",'select post marker channel image');
for j =1:Channel_userdif
    for i = 1:Num_grid
        file_omeopen = strcat(Channel_snap(j),'_',num2str(i),file_ome(end-28:end-11),num2str(i-
1,"%03d"),file_ome(end-7:end));
```

```
        path_omeopen = strcat(path_rawpre,'\',Channel_snap(j),'_',num2str(i),'\');
        MIJ.run("Bio-Formats", "open=[" + strcat(path_omeopen,file_omeopen) + "] color_mode=Default
concatenate_series open_all_series split_channels view=Hyperstack stack_order=XYCZT"); % open
"before" ome-tiff file
        IJ.runMacro("rename('"+Channel_snap(j)+"_"+num2str(i)+"');");
    end
    Conc = 'image'+string([1:Num_grid])+'=['+Channel_snap(j)+'_'+string([1:Num_grid])+']';
    MIJ.run("Concatenate...",strcat("title=",Channel_snap(j)+'pre'," ",join(Conc,1)));
end
for j =1:Channel_userdif
    for i = 1:Num_grid
        file_omeopen = strcat(Channel_snap(j),'_',num2str(i),file_ome2(end-28:end-11),num2str(i-
1,"%03d"),file_ome2(end-7:end));
        path_omeopen = strcat(path_rawpost,'\',Channel_snap(j),'_',num2str(i),'\');
        MIJ.run("Bio-Formats", "open=[" + strcat(path_omeopen,file_omeopen) + "] color_mode=Default
concatenate_series open_all_series split_channels view=Hyperstack stack_order=XYCZT"); % open
"after" ome-tiff file
        IJ.runMacro("rename('"+Channel_snap(j)+"_"+num2str(i)+"');");
    end
    Conc = 'image'+string([1:Num_grid])+'=['+Channel_snap(j)+'_'+string([1:Num_grid])+']';
    MIJ.run("Concatenate...",strcat("title=",Channel_snap(j)+'post'," ",join(Conc,1)));
end
serialNum = Num_posi;
reader=bfGetReader([path_ome,file_ome]);
omeMeta = reader.getMetadataStore();
PixelsizeC =    omeMeta.getPixelsSizeC(0);
PixelsizeX = omeMeta.getPixelsSizeX(0);
PixelsizeY = omeMeta.getPixelsSizeY(0);
PhysicalSize = double(omeMeta.getPixelsPhysicalSizeX(0).value());
sizeC = double(PixelsizeC.getNumberValue());
sizeX = double(PixelsizeX.getNumberValue());
sizeY = double(PixelsizeY.getNumberValue());
clear indx PixelsizeC PixelsizeX PixelsizeY;

% flat field correction and generation of image sequence for each channel
Mean_flat = zeros(1,size(file_flat,2));
MIJ.run("Set Measurements...", "area mean centroid stack redirect=None decimal=3");
for i = 1:size(file_flat,2)
    IJ.runMacro("open('"+path_flat+file_flat(i)+"');");
    MIJ.run("Select All");
    MIJ.run("Measure");
```

```
	flat_data = MIJ.getResultsTable();
	Mean_flat(i) = flat_data(2);
	IJ.runMacro("IJ.deleteRows(0,1);");
	IJ.runMacro("open('"+path_flat+file_background(i)+"');");
end
window_selected = MIJ.getListImages;
window_selected = string(window_selected);
window_selected(find(strncmp(window_selected,'Preview',7) == 1))= [];
temp_Channel = Channel_snap(:,ones(1,2))';
Channel = reshape(temp_Channel',numel(temp_Channel),1);
Channel = strcat(Channel,["-before" "-before" "-after" "-after"]');
% process "before" and "after" image
for no= 1:size(Channel,1)
	MIJ.run("Calculator Plus", "i1=["+window_selected(no)+"] i2=["+file_background(2-mod(no,2))+"]
operation=[Subtract: i2 = (i1-i2) x k1 + k2] k1=1 k2=0 create");
	MIJ.selectWindow(window_selected(no));
	IJ.runMacro('close();');
	MIJ.selectWindow("Result");
	MIJ.run("Properties...", "channels=1 slices=1 frames="+num2str(serialNum)+" unit=micron
pixel_width="+num2str(PhysicalSize)+" pixel_height="+num2str(PhysicalSize)+" voxel_depth=1.0000000
frame=[0.00 sec]");
	IJ.runMacro("rename('"+Channel(no)+"');");
	MIJ.run("Calculator Plus", "i1=["+Channel(no)+"] i2=["+file_flat(2-mod(no,2))+"] operation=[Divide: i2
= (i1/i2) x k1 + k2] k1="+Mean_flat(2-mod(no,2))+" k2=0 create");
	MIJ.selectWindow(Channel(no));
	IJ.runMacro('close();');
	MIJ.selectWindow("Result");
	MIJ.run("Properties...", "channels=1 slices=1 frames="+num2str(serialNum)+" unit=micron
pixel_width="+num2str(PhysicalSize)+" pixel_height="+num2str(PhysicalSize)+" voxel_depth=1.0000000
frame=[0.00 sec]");
	IJ.runMacro("rename('"+Channel(no)+"');");
	if no<=2
		path_save = dir + "/channel_" + Channel(no,1);
	else
		path_save = dir + "/channel_" + Channel(no,1);
	end
	mkdir(path_save);
	MIJ.run("Image Sequence... ", "format=TIFF name=stack_ save=[" + path_save + "]");
end;
for i = 1:size(file_flat,2)
	MIJ.selectWindow(file_flat(i));
```

```
    IJ.runMacro('close();');
    MIJ.selectWindow(file_background(i));
    IJ.runMacro('close();');
end
%% run stardist in a new FIJI window (for stardist screening)
% import post marker channel bg-substracted image sequence by bioformat, be
% sure that the t axis is set as the third axis. run stardist using default parameters and save the ROIs.
% [[NOTE]] you can reduce "persentile high" to 99 to eliminate the overexpressed cells

%% run cellpose in anaconda powershell prompt (for cellpose screening)
% conda activate cellpose
[file_cellpose,path_cellpose,indx] = uigetfile(dir_process + "\*.*",'select processed post-marker images for
segmentation');
system(['C:\Users\ZouOptics\anaconda3\envs\cellpose\python -m cellpose'...
    ' --dir ' [path_cellpose]...
    ' --pretrained_model cyto --chan 0 --use_gpu --fast_mode --save_png --no_npy --verbose']);
% open the processed image in a new FIJI by bioformat, then use
% "Plugins-BIOP-Image Analysis-ROIs-Label Image to ROIs" to get the ROI.
% saving the ROIs as .zip
%    --diameter 30
%% process stardist/cellpose returned result for confocal
[file_pro,path_pro,indx_pro] = uigetfile(dir_process + "\*.*",'select processed RoiSet');
MIJ.run("ROI Manager...");
IJ.runMacro("roiManager('reset');");
IJ.runMacro("roiManager('Open','"+replace(path_pro,'\','/') + file_pro+"');");
ResultsTable = cell(size(Channel,1),1);
IJ.runMacro('roiManager("count");');
MIJ.selectWindow("Log");
roiManagerSize = str2num(MIJ.getLog());
MIJ.run("Close" );
% IJ.runMacro("roiManager('save', '" + dir +"/libraryROI.zip"+ "');");
for no= 0:size(Channel,1)-1
    MIJ.selectWindow(Channel(no+1));
    %IJ.runMacro('roiManager("translate", 1, 0);');
    IJ.runMacro('roiManager("Measure");');
    ResultsTable(no+1,1) = {MIJ.getResultsTable()};
    IJ.runMacro("IJ.deleteRows(0, "+roiManagerSize+");");
end
MeanIntMarker_before = ResultsTable{1,1}(:,2);
MeanIntLib_before = ResultsTable{2,1}(:,2);
MeanIntMarker_after = ResultsTable{3,1}(:,2);
```

```
MeanIntLib_after = ResultsTable{4,1}(:,2);
Area = ResultsTable{1,1}(:,1);
Slice = ResultsTable{1,1}(:,5);
Xrev = ResultsTable{1,1}(:,3);
Yrev = ResultsTable{1,1}(:,4);
Xabs = zeros(roiManagerSize,1);
Yabs = zeros(roiManagerSize,1);
Zabs = zeros(roiManagerSize,1);
camera = string(omeMeta.getDetectorID(0,0));
if camera == "camera_confocal"
    theta = atan(16/(2454-512));       %camera confocal# num calculated from file G:\lc
data\20210227_LC_dish\dish test 2\CELL_8\\CELL_7_MMStack_1-Pos000_000.ome.tif
else
    theta = 0;    %camera widefield unknown
end
Xcor = Xrev-Xspot;
Ycor = Yrev-Yspot;
Xcor_rev = Xcor*cos(theta)+Ycor*sin(theta);
Ycor_rev = Ycor*cos(theta)-Xcor*sin(theta);
Xabs = position_slice(Slice,1)+Xcor_rev;
Yabs = position_slice(Slice,2)+Ycor_rev;
Zabs = position_slice(Slice,3);
Sensitivity = (MeanIntLib_after./MeanIntMarker_after-
MeanIntLib_before./MeanIntMarker_before)./(MeanIntLib_before./MeanIntMarker_before);
PAstate = ones(roiManagerSize,1); % 1 refers to unlabeled, 2 refers to labeled, 3 refers to discard
Index = [1:roiManagerSize]';
AnalysisTable =
table(Index,Area,Slice,Xrev,Yrev,Xcor_rev,Ycor_rev,Zabs,MeanIntMarker_before,MeanIntLib_before,Mea
nIntMarker_after,MeanIntLib_after,Sensitivity,PAstate);
clear Index Area Slice Xrev Yrev Xcor_rev Ycor_rev Zabs MeanIntMarker_before MeanIntLib_before
MeanIntMarker_after MeanIntLib_after Sensitivity PAstate ResultsTable Xabs Yabs Xcor Ycor;
rows = (AnalysisTable.Area<EdgeArea | AnalysisTable.Xrev<EdgeDis |
AnalysisTable.Xrev>sizeX*PhysicalSize-EdgeDis | AnalysisTable.Yrev<EdgeDis |
AnalysisTable.Yrev>sizeX*PhysicalSize-EdgeDis);
AnalysisTable = AnalysisTable(~rows,:);
%% find ROIgate (brightness)
figure();hold;
if para_sensi == 1
    Xscatter = AnalysisTable.MeanIntMarker_after;
    Yscatter = AnalysisTable.MeanIntLib_after; % use "MeanIntLib_after" for positive-going sensor
else
```

```
    Xscatter = AnalysisTable.MeanIntMarker_before;
    Yscatter = AnalysisTable.MeanIntLib_before; % use "MeanIntLib_before" for negative-going sensor
end
scatter(Xscatter,Yscatter,'Marker','.');
xlabel('Marker brightness');
set(gca, 'XScale', 'log');
set(gca, 'YScale', 'log');
title('select a gate');
ylabel('Library brightness');
gate = drawpolygon;
set(gate,'userdata',[Xscatter Yscatter]);
position_gate = customWait(gate);
in = inpolygon(log(Xscatter),log(Yscatter),log(position_gate(:,1)),log(position_gate(:,2)));
close(gcf);
figure();
plot(Xscatter(in),Yscatter(in),'r.',Xscatter(~in),Yscatter(~in),'b.');hold on
drawpolygon('Position',position_gate);
xlabel('Marker brightness');
set(gca, 'XScale', 'log');
set(gca, 'YScale', 'log');
title('select a gate');
ylabel('Library brightness');
title(gca,['Selected cell num:',num2str(sum(in))]);
path_Analysis = dir_process + "/Analysis";
mkdir(path_Analysis);
saveas(gca,path_Analysis+'/0 gating result.fig');
saveas(gca,path_Analysis+'/0 gating result.png');
BrightTable = AnalysisTable(in,:);
PATable = [];
clear in;
%% find ROIgate
% if you are evolving a sensor from a template without trying to reverse it, e.g. negative going NIR-GECO,
% positive going GRAB or negative going Ace2N-mNeon, run this section.
figure();hold;
if para_sensi == 1
    Xscatter = BrightTable.MeanIntLib_after;
else
    Xscatter = BrightTable.MeanIntLib_before; % use "MeanIntLib_before" for negative-going sensor
end
Yscatter = para_sensi*BrightTable.Sensitivity;
scatter(Xscatter,Yscatter,'Marker','.');
```

```
plot([min(Xscatter),max(Xscatter)],[para_sensi*tem_sensi,para_sensi*tem_sensi],'r--');
xlabel('Library brightness');
set(gca, 'XScale', 'log');
title('select a gate');
ylabel(strcat('Sensitivity (',sensi_txt,'\DeltaF/F)'));
gate = drawpolygon;
set(gate,'userdata',[Xscatter Yscatter]);
position_gate = customWait(gate);
in = inpolygon(log(Xscatter),Yscatter,log(position_gate(:,1)),position_gate(:,2));
close(gcf);
figure();
plot(Xscatter(in),Yscatter(in),'r.',Xscatter(~in),Yscatter(~in),'b.');hold on
plot([min(Xscatter),max(Xscatter)],[para_sensi*tem_sensi,para_sensi*tem_sensi],'r--');
drawpolygon('Position',position_gate);
xlabel('Library brightness');
set(gca, 'XScale', 'log');
title('select a gate');
ylabel(strcat('Sensitivity (',sensi_txt,'\DeltaF/F)'));
title(gca,['Selected cell num:',num2str(sum(in))]);
saveas(gca,path_Analysis+'/1-1 gating result.fig');
saveas(gca,path_Analysis+'/1-1 gating result.png');
PATable = BrightTable(in,:);
clear in;
%% generate position list
pl = gui.getPositionList(); % class org.micromanager.PositionList
pl.clearAllPositions();
XYStage_Label = mmc.getXYStageDevice();
PFSStage_Label = 'TIPFSOffset'; % for stage1 in A317 , use 'TIPFSOffset',for demo use 'Z'
if simulate == 1
    PFSStage_Label = 'Z';
end
msp = cell(size(PATable,1),1);
for i = 1:size(PATable,1)
PosiX = position_slice(PATable.Slice(i),1)+PATable.Xcor_rev(i);
PosiY = position_slice(PATable.Slice(i),2)+PATable.Ycor_rev(i);
PosiZ = position_slice(PATable.Slice(i),3);
msp(i,1) = org.micromanager.MultiStagePosition(XYStage_Label,PosiX,PosiY,PFSStage_Label,PosiZ);
msp{i,1}.setLabel(['cell_',num2str(PATable.Index(i))]);
pl.addPosition(msp{i,1});
end
%% save analysis data
```

```
save(path_Analysis+'/LibAnalysis.mat','AnalysisTable','BrightTable','PATable');
xlswrite(path_Analysis+'/LibAnalysis.xlsx',[AnalysisTable.Properties.VariableNames;table2cell(AnalysisTa
ble)],'AnalysisTable');
xlswrite(path_Analysis+'/LibAnalysis.xlsx',[PATable.Properties.VariableNames;table2cell(PATable)],'PATabl
e');
xlswrite(path_Analysis+'/LibAnalysis.xlsx',[BrightTable.Properties.VariableNames;table2cell(BrightTable)],'
BrightTable');
%%    run PA_manually
run PA_manually_sensor_screen;
%% functions
function pos = customWait(hROI)
title('adjust your gate');
% Listen for mouse clicks on the ROI
l = addlistener(hROI,'ROIClicked',@clickCallback);
l = addlistener(hROI,'ROIMoved',@movedCallback);
% Block program execution
uiwait;

% Remove listener
delete(l);

% Return the current position
pos = hROI.Position;

end

function clickCallback(~,evt)

if strcmp(evt.SelectionType,'double')
      uiresume;
end

end

function movedCallback(src,evt)
coordinate = get(src,'userdata');
in = inpolygon(log(coordinate(:,1)),coordinate(:,2),log(src.Position(:,1)),src.Position(:,2));
title(gca,['Selected cell num:',num2str(sum(in))]);
end
```

Method S2. MATLAB scripts for patch analysis of FR-GECIs, related to STAR Methods.

```
% patch analysis for NIR-GECO. this script is for finding AP peaks from DAQ
% data and calculate the Ca sensitivity of GECI.
% reference: stimulated AP multicycles
clear all; clc;
%% load and select
% Movie loading path
dir_process = 'D:\lc_data\20221018_LIN_CAV24\A7\cell2\';
dire = -1;    % dire = 1 means positive GECI; dire = -1 means negative GECI
peak_thres = 20;       % mV

pathname = uigetdir(dir_process,'Select patch-imaging subfolder');
listing = dir(pathname);
for i =1:size(listing,1)
    if listing(i).isdir == 1 && ~strcmp(listing(i).name,'.')    && ~strcmp(listing(i).name,'..') &&
exist([pathname '\' listing(i).name '\matlab variables.mat'])
        load ([pathname '\' listing(i).name '\matlab variables.mat']);
        break
    end
end
if exist([dir_process '\patch param.txt'])
    b = importdata([dir_process '\patch param.txt']);
    datab = b.data;Cm = datab(1,5);Rm = datab(1,4);Ra = datab(1,3);%pF

else
    Rm = 'N.A.';Cm = 'N.A.';Ra = 'N.A.';
end

if exist([pathname '\movie_info.txt'])
    c = importfile([pathname '\movie_info.txt']);
elseif exist([pathname '\movie.txt'])
    c = importfile([pathname '\movie.txt']);
end
movname = '\movie.bin';
ncol = c.DO(find(strncmp(c.laser,'nrow',4) == 1));              % x invert
```

```
nrow = c.DO(find(strncmp(c.laser,'ncol',4) == 1));                % y invert
camera_bias = c.DO(find(strncmp(c.laser,'Binning',7) == 1)).^2*100;    % background due to camera bias
(100 for bin 1x1)
dt_mov = c.DO(find(strncmp(c.laser,'Exposure',8) == 1));       % exposure time in millisecond (484 Hz)
Fs = samprate;
DAQname = '\movie_DAQ.txt';
dnsamp = Fs/(1000/dt_mov);              % downsampling rate = DAQ rate/camera rate
dnsamp = round(dnsamp);

% load DAQ data
tmp = importdata([pathname DAQname]);     % import data
data = tmp.data;                          % get array
Vm = data(:,2)*100;                       % Vm in millivolt, column vector
dt_daq = dt_mov/dnsamp;                     % DAQ dt in millisecond
t_daq = [0:length(Vm)-1]*dt_daq/10^3;             % DAQ time axis in second
a=importdata([pathname '\movie_DAQ.txt']);
data=a.data;
AI_scaled=data(:,1);
AI_10Vm=data(:,2)*100;
time=(1:length(AI_scaled)')'./Fs;
figure;
set(gcf,'outerposition',get(0,'screensize'));
plot(time,AI_scaled,time,AI_10Vm);
legend('AI\_scaled','Vm (mV)','Location','Northeast');
hold on
xlim=[0,max(time)];
ylim=[-70,-60];
xL=xlim;yL=ylim;
set(gca,'xtick',[0:5:max(time)])
box off
axis([xL yL])
axis tight
saveas(gca,[pathname '\0 waveform of AI.fig']);
saveas(gca,[pathname '\0 waveform of AI.png']);

%% loading the video movie
```

```
% load movie
fname = [pathname movname];
[mov, nframe] = readBinMov(fname, nrow, ncol);
mov = single(mov);img = mean(mov, 3);
img,camera_bias,max(img,[],'all'), 'Please select interested regions');
[~, intens_raw] = clicky(mov, img, 'select only 1 ROI, right click when done');
intens_rembkg = intens_raw(:,1)-intens_raw(:,2);
% select ROI for analysis
background = mean(intens_raw(:,size(intens_raw,2)));

saveas(gca,[pathname '\1 clicky analysis.fig']);
saveas(gca,[pathname '\1 clicky analysis.png']);
len = size(intens_raw,1);
t_mov = [0:(len-1)]*dt_mov/1000;         % time axis in second

%% dump kernel by left click on axes
h = figure('Name','dump kernel by left click on axes');
set(gcf,'Position',get(0,'ScreenSize'));
ax = {};dumpIndex = ones(1,cycles);
headPts_fluo = size(headPts,2)/dnsamp;       tailPts_fluo = size(tailPts,2)/dnsamp;
bleachPts =    2*samprate/dnsamp;       cyclePts = (hiPts+lowPts)/dnsamp;
for i = 1:cycles
      kernel_Vm(1:hiPts+lowPts,i) = AI_10Vm(size(headPts,2)+(i-1)*(hiPts+lowPts)-
round(2*samprate)+1:size(headPts,2)+i*(hiPts+lowPts)-round(2*samprate));
      kernel_Fluo_dump(1:cyclePts,i) = intens_raw(headPts_fluo+(i-1)*cyclePts-
round(bleachPts)+1:headPts_fluo+i*cyclePts-round(bleachPts));
      ax{2*i-1} = subplot(cycles,2,2*i-1);       plot([0:hiPts+lowPts-1],kernel_Vm(:,i));
      ax{2*i} = subplot(cycles,2,2*i);       plot([0:cyclePts-1],kernel_Fluo_dump(:,i));
      if i<cycles
            set(ax{2*i-1},'xticklabel',[]);       set(ax{2*i},'xticklabel',[]);
      end

end
hold on
set(h.Children,'buttondownfcn',{@buttondownfcn,cycles,ax,pathname});              % assign function to
gca
```

```
saveas(h,[pathname '\2 dumped kernel.fig']);
saveas(h,[pathname '\2 dumped kernel.png']);
%% photobleaching correction
% exponential fitting on remained kernels
intens_rembkg_norm = intens_rembkg./mean(intens_rembkg(1:round(headPts_fluo/100)));
plot(t_mov,intens_rembkg_norm');
F = @(x,xdata) x(1).*exp(-x(2).*xdata) + x(3).*exp(-x(4).*xdata);
x0 = [0.9 0.004 0.1 0.0001];
period_dataPts = [ones(1,round(bleachPts)), zeros(1,cyclePts-round(bleachPts))];
period_dataPts_false = [zeros(1,round(bleachPts)), zeros(1,cyclePts-round(bleachPts))];
% Cycle each period to give steps
dataPts = [ones(1,round(headPts_fluo)-round(bleachPts))];
for i = 1:cycles
if logical(dumpIndex(i))
    dataPts = [dataPts period_dataPts];
else
    dataPts = [dataPts period_dataPts_false];
end
end
dataPts = [dataPts zeros(1,round(bleachPts)) ones(1,round(tailPts_fluo))];
xdata = t_mov(logical(dataPts)); ydata = double(intens_rembkg_norm(logical(dataPts))');
x = lsqcurvefit(F,x0,xdata,ydata)
hold on
plot(t_mov,F(x,t_mov));
hold off
saveas(gca,[pathname '\3_1 photobleaching correction.fig']);
saveas(gca,[pathname '\3_1 photobleaching correction.png']);
close(gcf);
intens_corr = intens_rembkg_norm./F(x,t_mov)';
plot(t_mov,intens_rembkg_norm./F(x,t_mov)');
saveas(gca,[pathname '\3_2 trace after photobleaching correction.fig']);
saveas(gca,[pathname '\3_2 trace after photobleaching correction.png']);
%% calculate Ca deltaF/F versus AP number
% generate kernel
figure()
tkernel_Vm = repmat([0:hiPts+lowPts-1]*dt_daq/10^3,cycles,1)';
```

```
tkernel_Fluo = repmat([0:cyclePts-1]*dt_mov/10^3,cycles,1)';
locs_Vpeak = {};
Num_Vpeak = [];
for i = 1:cycles
kernel_Fluo(1:cyclePts,i) = intens_corr(headPts_fluo+(i-1)*cyclePts-
round(bleachPts)+1:headPts_fluo+i*cyclePts-round(bleachPts));
[~,locs] = findpeaks(kernel_Vm(:,i),'MinPeakHeight',peak_thres);
Num_Vpeak=[Num_Vpeak size(locs,1)];
locs_Vpeak{i} = locs;

end
kernel_Fluo_smo = smoothdata(kernel_Fluo,1,'sgolay',30);
if dire == 1
    FluoPeak = max(kernel_Fluo_smo,[],1);
elseif dire == -1
    FluoPeak = min(kernel_Fluo_smo,[],1);
end
FluoSteady = mean(kernel_Fluo_smo(1:round(bleachPts),:),1);
SensiPeak = (FluoPeak-FluoSteady)./FluoSteady;
FluoPeak_half = (FluoPeak+FluoSteady)./2;
kernel_Fluo_norm = kernel_Fluo./FluoSteady;
kernel_Fluo_smo_norm = kernel_Fluo_smo./FluoSteady;
SNR = abs(FluoPeak-FluoSteady)./std(kernel_Fluo(round(bleachPts)-
round(bleachPts*0.25):round(bleachPts),:),1);
thalf_rise = [];
thalf_decay = [];
for i = 1:cycles
    if dumpIndex(i) == 1
        subplot(cycles,1,i);
        plot(tkernel_Fluo(:,i),kernel_Fluo_norm(:,i),tkernel_Fluo(:,i),kernel_Fluo_smo_norm(:,i));
        hold on
        Peak_Pt = find(kernel_Fluo_smo(:,i)==FluoPeak(i));
        if dire == -1
            thalf_rise_Pt = min(find(kernel_Fluo_smo(round(bleachPts)+1:Peak_Pt(1),i) <=
FluoPeak_half(i)));
            thalf_decay_Pt = min(find(kernel_Fluo_smo(Peak_Pt(1):end,i) >= FluoPeak_half(i)));
```

```
            else
                thalf_rise_Pt = min(find(kernel_Fluo_smo(round(bleachPts)+1:Peak_Pt(1),i) >=
FluoPeak_half(i)));
                thalf_decay_Pt = min(find(kernel_Fluo_smo(Peak_Pt(1):end,i) <= FluoPeak_half(i)));
            end
            thalf_rise = [thalf_rise thalf_rise_Pt*dt_mov];
            thalf_decay = [thalf_decay thalf_decay_Pt*dt_mov];

plot(tkernel_Fluo(round(bleachPts)+thalf_rise_Pt,i),kernel_Fluo_smo_norm(round(bleachPts)+thalf_rise_
Pt,i),'r.');
            plot(tkernel_Fluo(Peak_Pt(1)+thalf_decay_Pt-
1,i),kernel_Fluo_smo_norm(Peak_Pt(1)+thalf_decay_Pt,i),'r.');
            hold off
        else
            thalf_rise = [thalf_rise NaN];
            thalf_decay = [thalf_decay NaN];
        end
end
saveas(gca,[pathname '\4 norm kernel with thalf.fig']);
saveas(gca,[pathname '\4 norm kernel with thalf.png']);
figure()
plot(tkernel_Fluo(:,logical(dumpIndex)),kernel_Fluo_norm(:,logical(dumpIndex)));
saveas(gca,[pathname '\5-1 norm kernel stack.fig']);
saveas(gca,[pathname '\5-1 norm kernel stack.png']);
close(gcf)
figure()
plot(tkernel_Fluo(:,logical(dumpIndex)),kernel_Fluo_smo_norm(:,logical(dumpIndex)));
saveas(gca,[pathname '\5-1 norm smooth kernel stack.fig']);
saveas(gca,[pathname '\5-1 norm smooth kernel stack.png']);
close(gcf)
%%
Num_Vpeak_save = Num_Vpeak(logical(dumpIndex));
SensiPeak_save = SensiPeak(logical(dumpIndex));
thalf_rise_save = thalf_rise(logical(dumpIndex));
thalf_decay_save = thalf_decay(logical(dumpIndex));
SNR_save = SNR(logical(dumpIndex));
```

```
save([pathname
'\analysis.mat'],'AI_10Vm','dumpIndex','FluoPeak','FluoSteady','intens_rembkg','intens_corr','kernel_Fluo','
kernel_Fluo_norm','kernel_Fluo_smo','kernel_Fluo_smo_norm','kernel_Vm','tkernel_Fluo','tkernel_Vm');
xlswrite([pathname '\analysis.xlsx'],{'AP No.','Peak
response','t_half_rise','t_half_decay','SNR','Ra','Rm','Cm'},'Raw','A1');
xlswrite([pathname '\analysis.xlsx'],Num_Vpeak_save','Raw','A2');
xlswrite([pathname '\analysis.xlsx'],SensiPeak_save','Raw','B2');
xlswrite([pathname '\analysis.xlsx'],thalf_rise_save','Raw','C2');
xlswrite([pathname '\analysis.xlsx'],thalf_decay_save','Raw','D2');
xlswrite([pathname '\analysis.xlsx'],SNR_save','Raw','E2');
xlswrite([pathname '\analysis.xlsx'],Ra,'Raw','F2');
xlswrite([pathname '\analysis.xlsx'],Rm,'Raw','G2');
xlswrite([pathname '\analysis.xlsx'],Cm,'Raw','H2');

Num_Vpeak_uni = unique(Num_Vpeak(logical(dumpIndex)));
for i = 1:size(Num_Vpeak_uni,2)
    lo = (Num_Vpeak_save == Num_Vpeak_uni(i));
    SensiPeak_uni(i) = mean(SensiPeak_save(lo));
    thalf_rise_uni(i) = mean(thalf_rise_save(lo));
    thalf_decay_uni(i) = mean(thalf_decay_save(lo));
    SNR_uni(i) = mean(SNR_save(lo));
end
xlswrite([pathname '\analysis.xlsx'],{'AP No.','Peak
response','t_half_rise','t_half_decay','SNR'},'Average','A1');
xlswrite([pathname '\analysis.xlsx'],Num_Vpeak_uni','Average','A2');
xlswrite([pathname '\analysis.xlsx'],SensiPeak_uni','Average','B2');
xlswrite([pathname '\analysis.xlsx'],thalf_rise_uni','Average','C2');
xlswrite([pathname '\analysis.xlsx'],thalf_decay_uni','Average','D2');
xlswrite([pathname '\analysis.xlsx'],SNR_uni','Average','E2');
%% functions
function buttondownfcn(hobj,~,cycles,ax,pathname)
dumpIndex = evalin('base','dumpIndex');
for i = 1:cycles
    if isequal(ax{2*i-1}.Position,get(hobj,'Position')) | isequal(ax{2*i}.Position,get(hobj,'Position'))
        dumpIndex(i) = 0;
```

```
            title(ax{2*i-1},'dumped kernel');
            title(ax{2*i},'dumped kernel');
        end
end
% set(hobj.Parent,'userdata',dumpIndex);          % add this line insdie the function
disp(dumpIndex);
assignin('base','dumpIndex',dumpIndex);
saveas(hobj.Parent,[pathname '\2 dumped kernel.fig']);
saveas(hobj.Parent,[pathname '\2 dumped kernel.png']);
end
```