Supplementary Information for

Machine Learning Recognition of Protein Secondary Structures based on Two-Dimensional Spectroscopic Descriptors

Hao Ren[1], Qian Zhang[1], Zhengjie Wang[1], Guozhen Zhang[2], Hongzhang Liu[1], Wenyue Guo[1], Shaul Mukamel[3,*], Jun Jiang[2,*]

1. School of Materials Science and Engineering, China University of Petroleum (East China), Qingdao 266580, Shandong, China
2. School of Chemistry and Materials Science, University of Science and Technology of China, Hefei 230026, Anhui, China
3. Department of Chemistry and Physics & Astronomy, University of California, Irvine, California, 92697, United States

Shaul Mukamel
Email: smukamel@uci.edu

Jun Jiang
Email: jiangj1@ustc.edu.cn

This PDF file includes:

Supplementary text
Figures S1 to S3
Tables S1 to S4

Other supplementary materials for this manuscript include the following:

S1. Simulation of 2DUV spectra

2DUV photon echo spectra were simulated using four coherent broad band ultrafast ultraviolet pulses, with wavevectors $\boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{k}_3,$ and $\boldsymbol{k}_4$. The signals are detected in the direction $\boldsymbol{k}_4 = -\boldsymbol{k}_1 + \boldsymbol{k}_2 + \boldsymbol{k}_3$ with varying time delays t_1, t_2 and t_3. The second time delay t_2 was set to zero, so that the photon echo signals depend on t_1 and t_3. 2D frequency-domain signals were then obtained by performing 2D Fourier transforms: $(t_1, t_3) \rightarrow (\Omega_1, \Omega_3)$. All pulses have the same linear polarization. We used Gaussian pulses centered at 52000 cm^{-1} (~190 nm) with a full width at half maximum (FWHM) of 3000 cm^{-1}. Signals in the frequency range 42000-58000 cm^{-1} (~238-172 nm) were calculated to generate the linear absorption (LA) and 2DUV spectra.

S2. Details of the CNN classifier

1. 1D CNN for LA and CD processing

The 1D CNN models for LA and CD processing adopt the same architecture (Fig. S1): start from an input layer, followed by N convolutional modules and a drop out layer, then a fully-connected model, a drop out layer and finally a softmax output layer. The number of convolutional modules and the number of filters/channels therein were optimized with the grid search method. Each group of the CNN filters was followed by a max pooling layer with pooling window size varies from 2 to 20, which was also optimized with grid search.

The input layer has the dimension of 1601×1, corresponds to the intensity sequence in the range 42000~58000 cm^{-1} with the step size of 100 cm^{-1}. The drop out rates were set to 0.25 for all the drop-out layers. The options of hyperparameters optimized with the grid search method were listed in Table S1.

2. 2D CNN for 2DUV processing

The 2D CNN models for 2DUV processing (**Fig. S2**) consists of an input layer, followed by N convolutional modules and a drop out layer, then a fully-connected model, a drop out layer and finally a softmax output layer. The number of convolutional modules and the number of filters/channels therein were optimized with the grid search method. Each group of the CNN filters was followed by a max pooling layer with pooling window sizes varies from 2 to 20, which was also optimized with grid search.

The input layer has the dimension of 161×161, corresponds to the 2DUV intensity distribution with respect to Ω_1 and Ω_3 in the range 42000~58000 cm^{-1} with the step size of 1000 cm^{-1}. The drop out rates were set to 0.55 for all the drop-out

layers. The options of hyperparameters optimized with the grid search method were listed in **Table S2**.

The 1D and 2D CNN models were trained by using the backpropagation algorithm with the adaptive moment estimation (Adam) optimizer. The layers were initialized with a Glorot uniform initializer; cross entropy loss was used as the loss function. We also applied early stopping to prevent overfitting. The training of the CNN was accelerated by employing four NVIDIA GeForce GTX-2080 Ti GPUs on a dual Xeon Silver 4110 workstation.

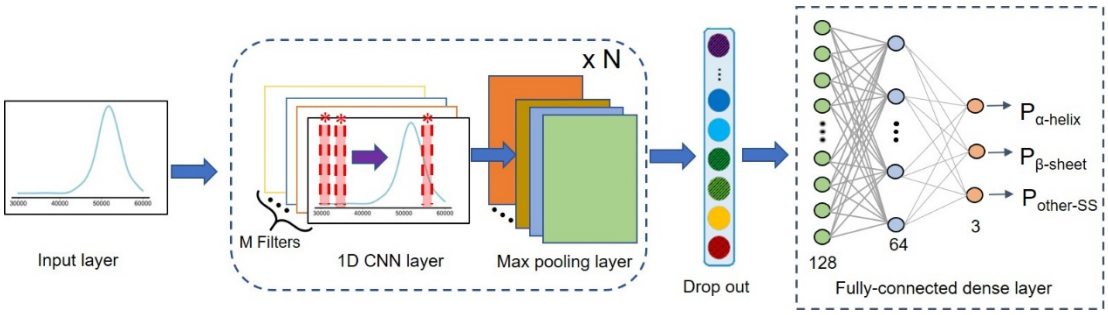

Figure S1. Scheme of the architecture of the 1DCNN model.

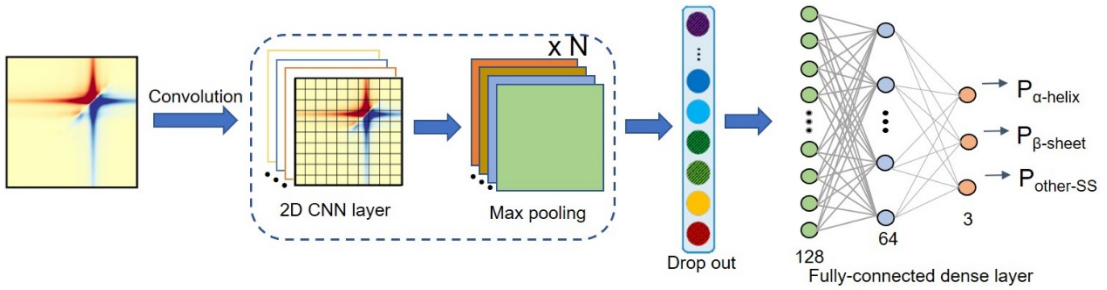

Figure S2. Scheme of the architecture of the 2D CNN model.

Table S1. Options of hyperparameters of the 1D CNN model optimized with the grid search method.

Hyperparameter	**values**
Number of convolution layers	1, 2, 3
Number of filters	32, 64, 128, 256, 512
max pooling window size	range from 2 to 20, step is 1
Learning rate	0.01, 0.001, 0.002, 0.004, 0.008, 0.0001, 0.0004, 0.0008
Batch size	32, 64, 128
Size of the fully-connected layers	32, 64, 128, 256, 512
Dropout rate	Range 0.1 to 0.5
Epochs	Early stopping, patience 5

Table S2. Options of hyperparameters of the 2D CNN model optimized with the grid search method.

Hyperparameter	**values**
Number of convolution modules	1, 2, 3
Number of filters	32, 64, 128, 256, 512
max pooling window size	range from 2 to 20, step is 2
Learning rate	0.01, 0.001, 0.002, 0.004, 0.008, 0.0001, 0.0004, 0.0008
Batch size	32, 64, 128
Size of the fully-connected layers	32, 64, 128, 256, 512
Dropout rate	Range 0.1 to 0.5

S3. Comparison between various recognition models

We had further trained and tested these models on the same datasets used for the CNN model. The incorrect recognitions by these models are presented in Table S3. We found that the recognition performance of these traditional models is significantly lower than that of the CNN model. Specifically, the KNN models achieved the best total error score 163 (out of 17599) when using hyperparameter $k = 1$ (only nearest neighbor); for SVM models, a linear kernel performs much better than radial basis function (rbf) kernel, and using polynomial kernel leads to much worse performance (7687 errors out of 17599); RF and FCNN also work well on this dataset, performs on par with the SVM model using a linear kernel. In conclusion, due to its powerful feature extraction ability, the CNN model out-performed all other traditional models.

Table S3. Numbers of incorrect recognitions for each structural category produced by different machine learning methods applied on the same datasets.

Methods	KNN(k=5)	KNN(K=3)	KNN(K=1)	SVM(linear)	SVM(poly)	SVM(rbf)	RF	FCNN	CNN
α-helix	94	93	95	4	548	62	10	4	1
β-sheet	77	75	64	7	1996	79	4	7	0
other-SS	3	1	4	2	5143	69	8	1	0
Total	174	169	163	13	7687	210	22	12	1

S4 Secondary structure discrimination accuracies of the pre-trained CNN models

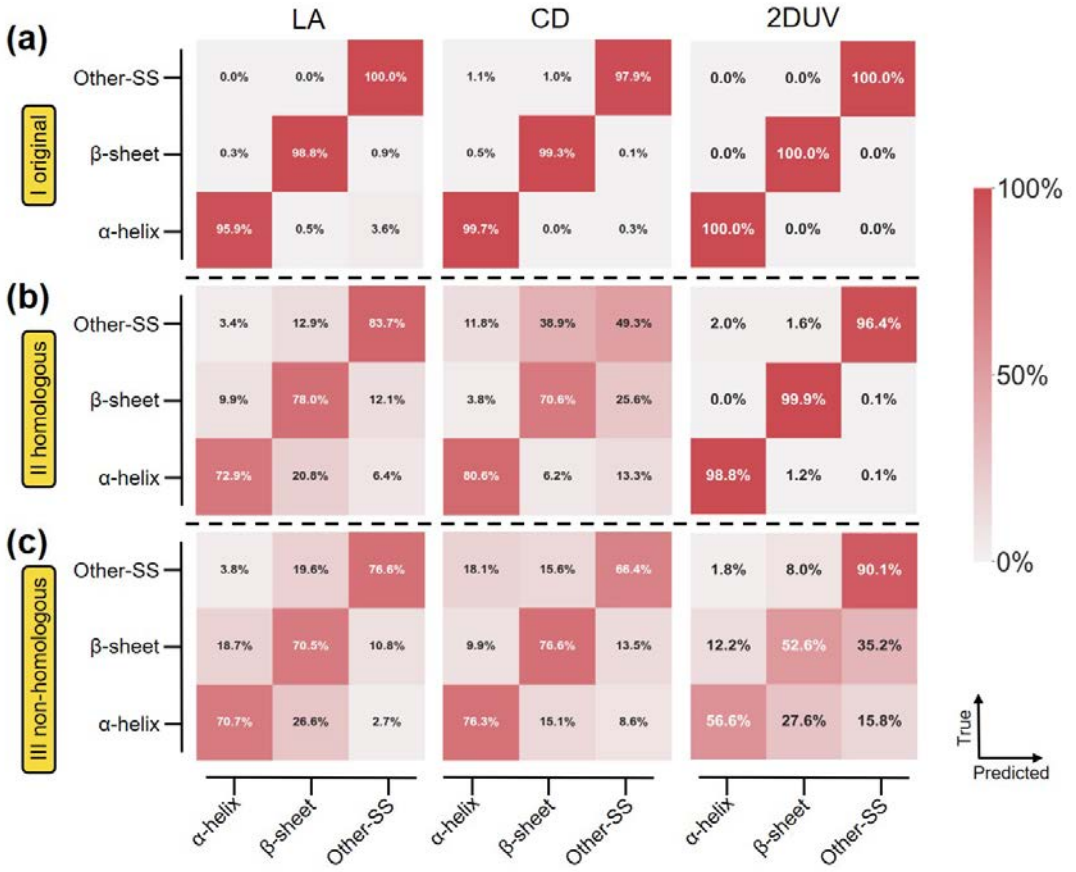

Figure S3. Confusion matrices of the pre-trained CNN models to recognize secondary structures of protein segments of (a) the original set I, (b) the homologous set II, and (c) the non-homologous set III. The vertical and horizontal axes represent true and model predicted secondary structures, respectively.

S5. Details of molecular dynamics simulations of proteins.

For each protein studied in this work, the X-ray/NMR crystal structure taken from the RCSB protein data bank (PDB) was used to initialize a molecular dynamics (MD) simulation performed by using the Gromacs package. Taking the BH protein as example, we put the protein molecule in a $9 \times 9 \times 9$ nm^3 cubic box with 21635 TIP3P water molecules. Following 1000 steps of energy minimization, a 200 ps equilibration with constant NVT at 300 K was performed. A 200 ps constant NPT equilibration and a 200 ps constant NVT equilibration were followed. The 4ns production equilibration was then performed with 1 fs time step. Snapshots were harvested every 1000 fs along the production MD trajectory to avoid structural coherence.

S6. Proteins used to construct the non-homologous dataset

All proteins are recorded with their PDB IDs. All PDB structures were directly downloaded from the RCSB protein data bank, followed by solvation in water, energy minimization, and NVT equilibration at 300 K. Peptide segments were then extracted from the equilibrated structures in the same way we prepare the BH and LL dataset.

Table S4. PDB IDs of proteins used to construct the non-homologous dataset.

1a00	1a01	1a0n	1a0u	1a2i	1a2s	1a3o	1a4f	1a6g	1a6m
1aby	1afp	1ah6	1ah8	1aj9	1amx	1anb	1aox	1aox	1ash
1ax8	1ayj	1b0b	1b1a	1b86	1b9q	1bbb	1bf8	1bij	1bk8
1bkv	1bpr	1bpr	1bsn	1buw	1buy	1bvc	1bvd	1c3g	1c40
1c89	1cbl	1ceu	1cg5	1cg8	1cgd	1ch4	1cjq	1ck2	1ck7
1ckr	1clg	1cmy	1cn4	1co9	1coh	1cpz	1d2p	1d5d	1d9a
1d9i	1dbd	1ddr	1dg4	1dgf	1dgh	1dke	1dkg	1dkx	1dky
1dlw	1dm1	1dox	1dxu	1dy2	1dy2	1dzi	1dzi	1e1g	1ebt
1ecd	1eer	1ey4	1ezu	1f4j	1f6h	1faw	1fcs	1fdh	1fdm
1fhj	1flp	1fm1	1fsz	1fuj	1fy9	1g08	1g0a	1g3j	1gcv
1gd4	1ght	1gjn	1gr3	1gr3	1gvl	1gxd	1gzx	1h1x	1h4u
1h6w	1h7c	1hab	1hba	1hbg	1hbh	1hbs	1hco	1hda	1hga
1hgb	1hgc	1hjn	1hk7	1hx1	1hyl	1hze	1i6z	1i7x	1ibe
1iox	1ird	1ivt	1iwh	1ix5	1j14	1j3z	1j52	1j7w	1j7y
1jb3	1jbk	1jf3	1jj9	1jon	1jvx	1jwn	1jy7	1jzk	1jzl
1jzm	1k0v	1k0y	1k9o	1kd2	1kfr	1khy	1kid	1kiu	1kke
1koe	1kr7	1l2y	1l8z	1la1	1les	1lfl	1lfq	1lft	1lfv
1li1	1m3d	1m9p	1mba	1mbd	1mbn	1mbo	1mbs	1mdi	1mgn
1mhp	1mko	1moh	1mol	1mwb	1myh	1myi	1myk	1mym	1myz
1mz0	1n9x	1nej	1nih	1npf	1npg	1nqp	1nwi	1nwn	1o1i
1o1k	1o1n	1o4w	1o91	1ocy	1oo4	1oqv	1ory	1p9h	1pbx
1pft	1pk6	1pmb	1pt6	1q5l	1q7d	1qc5	1qi8	1qiu	1qld
1qpw	1qqw	1qsd	1qun	1qvr	1qwx	1qxd	1r1x	1r1y	1rbw
1roc	1rps	1rtx	1rvw	1s21	1s5y	1s69	1s6a	1s85	1sb6
1sdk	1sdl	1shr	1si4	1slu	1spg	1ss3	1ss8	1swm	1t08
1t60	1t61	1t7s	1tey	1thb	1tjc	1tnw	1tpm	1tr8	1ttw
1tu9	1u5m	1u7s	1u97	1uiw	1ulo	1umk	1us7	1usu	1uvy
1uw3	1ux8	1uym	1uz2	1v4u	1v4w	1v4x	1v8x	1v9q	1vre
1w09	1w0a	1w0b	1wg3	1wvp	1wxr	1wxv	1x3b	1x3k	1x46
1x9f	1xu0	1xuc	1xxt	1xye	1xz2	1xzy	1y01	1y09	1y2s
1y4p	1y5j	1y8h	1y8i	1yca	1ydz	1yeo	1yeq	1ygf	1yhu
1yie	1yjp	1ykt	1ymb	1you	1yut	1yvq	1yvt	1yzb	1yzi
1z2g	1z8u	1zav	1ze3	1zrj	1ztq	1zwh	2a3g	2aa1	2adn
2akp	2arw	2av0	2b7h	2beg	2bmm	2bpr	2brc	2bre	2bsf

2bw9	2bwh	2c0k	2c0x	2cg9	2cge	2cmm	2cpb	2cu9	2d1n
2d2m	2d3e	2d5x	2d5z	2d60	2d6c	2dhb	2dk1	2dkm	2dkm
2dn1	2dn2	2dn3	2dxm	2e2d	2e2y	2e3m	2e3o	2e3r	2e8j
2ech	2eku	2evp	2f2n	2f42	2f68	2f6a	2fam	2fc6	2fcw
2frf	2frj	2fse	2fse	2fxs	2g0s	2g12	2g16	2gtl	2gtv
2h35	2h8d	2h8f	2hbc	2hbd	2hbf	2hbg	2hbs	2hco	2hhb
2hhd	2hhe	2hp8	2hue	2hz1	2idc	2iij	2in4	2iw2	2iws
2j61	2j7l	2jhh	2jhi	2jho	2jjc	2kb0	2kc5	2kco	2kgl
2kho	2kji	2knx	2ksc	2l6l	2lhb	2lhk	2lkv	2lll	2lll
2llp	2lm1	2ltb	2lwp	2lyj	2lyk	2lyl	2lyp	2lyq	2lyr
2lys	2m0m	2m1n	2m3e	2m6z	2m8s	2mb5	2mbw	2mgo	2miq
2mj5	2mye	2myj	2mze	2mzi	2n3j	2n4g	2n71	2n8r	2nb0
2nd2	2nd3	2nd5	2npl	2nrl	2nsa	2nsb	2nsr	2nx0	2o5l
2o5q	2o5s	2ohb	2oj5	2okm	2okn	2pei	2peo	2peq	2pgh
2qg2	2qht	2qif	2qld	2qls	2qsp	2qss	2qu0	2r1h	2r80
2r9y	2rao	2rk6	2rpj	2seb	2seb	2tgf	2uur	2uwj	2v15
2v1e	2v1f	2v1i	2v1k	2v53	2v7y	2vlx	2vly	2vw5	2vwc
2vyw	2vyy	2w0g	2w60	2w6v	2w6w	2w72	2wep	2wnp	2ww7
2xd6	2xi6	2xif	2xil	2xj6	2xki	2xx4	2y1z	2y6y	2yge
2yjm	2yob	2yrs	2yuh	2z1p	2z44	2z46	2z6s	2z6t	2z85
2z9y	2z9z	2zlv	2zlw	2zlx	2zsp	2zsq	2zss	2zsy	2zwh
2zwj	2zyp	3a0g	3a2g	3a59	3aeh	3aei	3ak5	3aq5	3ase
3asw	3asw	3b72	3b75	3bj1	3bwu	3c11	3ciu	3d17	3d1k
3d3r	3d7o	3dhr	3dll	3dpa	3dpo	3dpq	3dut	3eda	3ejh
3elm	3eok	3eu1	3ewo	3ewq	3f71	3fh9	3fp8	3fs4	3fzh
3fzk	3gkv	3gla	3gln	3gou	3gqg	3gqp	3gt6	3gys	3h0x
3h3q	3h3t	3hc9	3hf4	3hrv	3ia3	3ic0	3ic2	3ipn	3iuc
3k1h	3k8b	3kek	3l1e	3ld1	3ldl	3ldn	3ldo	3ldp	3ldq
3lfo	3ljz	3lqd	3lr7	3lw2	3m0b	3m38	3m3b	3m6c	3mba
3mjp	3mju	3mvf	3n3e	3n3f	3nl7	3nml	3o2x	3o39	3odq
3ofg	3ofh	3ogb	3oly	3osx	3ovu	3p46	3pel	3pg0	3pi8
3pi9	3pr9	3q9q	3qc7	3qje	3ql1	3qle	3qm5	3qzl	3qzm
3qzn	3qzo	3rik	3rjr	3rt5	3rtl	3rur	3s48	3s4u	3s5c
3s5h	3s5k	3sdh	3sdn	3sea	3sz7	3szk	3tee	3tfb	3tgf
3tm3	3tnj	3tnu	3tvc	3tvn	3uhi	3uj1	3umm	3ut2	3uyx
3v03	3v2v	3vfe	3vm5	3vm9	3vnd	3vnw	3vqk	3vql	3vqm
3vz6	3vz9	3w6l	3wai	3wft	3whm	3wi8	3wtg	3wv1	3wvl
3wyo	3zgh	3zgi	3zha	3zhc	3zhd	3zhk	3zhl	3zri	4a0q
4a7b	4ait	4aix	4aiz	4aj0	4am9	4ani	4asv	4au2	4au3
4b2t	4b9q	4bb2	4bj3	4bkl	4bkl	4bnr	4bpy	4bt9	4c0n
4c44	4cpg	4ctd	4cud	4cue	4cuf	4d0e	4d2u	4d3e	4d7y
4d8n	4dc5	4dc7	4df3	4dou	4dwf	4eew	4eo5	4ezn	4ezo
4ezp	4ezr	4ezw	4ezx	4f01	4f1z	4f3j	4f4o	4f68	4fc3

4fct	4fcw	4fei	4fum	4fup	4fvl	4fwz	4g6t	4gf3	4gr7
4h32	4hrr	4hrt	4hse	4hwc	4i0c	4i0y	4i1e	4i2s	4i37
4i3n	4i96	4igi	4igi	4ihk	4ij2	4j5m	4ja7	4ja9	4jb0
4jb2	4jnf	4jsd	4jso	4k07	4k5q	4k6g	4k6h	4k6k	4kjt
4kqt	4l2a	4l2c	4l2d	4lj6	4lja	4ll6	4lnz	4lx2	4m4b
4m56	4m8u	4ma7	4mbn	4mjh	4mkf	4mkg	4mkh	4mpr	4mqk
4mtc	4mth	4n79	4n7p	4n8w	4ni0	4nla	4nsm	4nwe	4nwh
4nyt	4o4t	4o4z	4odk	4odn	4odp	4of9	4oj0	4ood	4ow4
4ox0	4pnj	4pqb	4qby	4qyw	4r1e	4rmb	4rmb	4rrp	4rx9
4rzk	4tql	4tt0	4tyu	4u3h	4u5t	4u8u	4uos	4uot	4uox
4uoy	4urg	4urq	4urs	4uzv	4w68	4w70	4w81	4w94	4wbr
4wch	4wjg	4wt3	4wuy	4x86	4xif	4xif	4xs0	4y00	4yu3
4yu4	4yxl	4z3v	4zgg	4zly	4zry	5ab8	5aks	5ao6	5aqg
5aqi	5aqo	5aqt	5azq	5b06	5b5o	5b85	5boy	5bx0	5c6y
5cdk	5ce5	5cjb	5cjb	5cmv	5cn5	5cnc	5ctd	5cti	5cuz
5cva	5cvb	5d5r	5dut	5e3x	5e83	5e84	5e85	5eii	5eiv
5f2r	5ffo	5fqd	5fwl	5fwp	5ghu	5gw4	5gw5	5h22	5hba
5hgj	5hj2	5hly	5hq3	5hu6	5hy8	5i4w	5iat	5iax	5icu
5iks	5ilm	5ilp	5ilr	5j3p	5j3s	5j3z	5j7n	5jdo	5jg9
5jhi	5ji4	5jjd	5jom	5jui	5k31	5ker	5ki0	5kkk	5krw
5ksi	5ksj	5kvn	5kwx	5kwz	5kx0	5kx1	5kx2	5m3l	5m4g
5m4j	5m4l	5mba	5mby	5mc1	5mu0	5mu0	5mv3	5mzu	5n30
5n4h	5nax	5nay	5naz	5ni1	5nir	5njx	5nro	5nx3	5o4p
5obu	5ocx	5ocx	5ofo	5oj9	5oja	5omp	5omy	5opw	5opx
5ou8	5ou8	5ou9	5owi	5owj	5sv3	5sv7	5thp	5tu7	5tu8
5tu9	5u2l	5u2u	5ucb	5ucu	5ue2	5ue5	5uea	5uek	5urc
5ut7	5ut9	5uwk	5uyx	5v4m	5v4n	5vmm	5vpn	5vqp	5vsx
5vy8	5vy9	5vzn	5vzo	5vzp	5vzq	5w0s	5wez	5wo1	5wog
5wy9	5wyo	5x2r	5x2s	5xef	5xi9	5xir	5xkv	5xl0	5y45
5yan	5ycg	5yp8	5ypb	5ypg	5yup	5yzf	5z5o	5z8i	5zba
5zdi	5zfo	5zhb	5zui	5zyg	5zyk	5zz0	5zzf	5zzg	5zzt
5zzy	6a06	6a0v	6a0y	6a19	6a1w	6a23	6a2u	6a32	6a39
6a3c	6a4n	6ahf	6ait	6as9	6asy	6axb	6b99	6bb5	6bie
6bin	6bin	6bjr	6bnr	6bp9	6bwu	6cd2	6cf0	6cii	6cn8
6cqg	6cqv	6d45	6d6s	6dfm	6dju	6dl9	6dnm	6drq	6dtc
6e0f	6e0g	6e14	6e15	6e2j	6e2j	6e7g	6e7h	6ec0	6ec0
6ed3	6eof	6ewn	6f0f	6f0y	6f17	6f18	6f25	6fqf	6fse
6ftk	6fzw	6g5a	6g5b	6g5t	6gzd	6h2p	6h2q	6hal	6hbi
6hbw	6hfo	6hg7	6hv2	6ihx	6ii1	6j0a	6j81	6jp1	6k01
6m8f	6mv0	6n02	6n8v	6n8z	6nbc	6nbd	6nd8	6nd8	6ndh
6o5v	6o69	6og3	6owx	6p7s	6prq	6qff	6qfh	6qh9	6qi8
6rwt	6u3r	7hsc	7pck						